# Table of Contents

1  Simulation Modeling with Simul8 ............................................................... 4  
   1.1  What is simulation? .............................................................. 5  
   1.2  Let’s Start Simul8 ............................................................... 12  
   1.3  A Simple Model ............................................................... 15  
   1.4  Simul8’s Building Blocks ...................................................... 18  
   1.5  Building the Model in Simul8 ............................................... 24  
   1.6  Running the Model .............................................................. 29  
   1.7  Analyzing the Results ......................................................... 32  

2  Intermediate Modeling in Simul8 .............................................................. 54  
   2.1  The Call Center .................................................................. 55  
   2.2  Labels .............................................................................. 62  
   2.3  Sub-Windows .................................................................... 65  
   2.4  Time Dependent Arrivals ...................................................... 66  
   2.5  Routing Out ...................................................................... 67  
   2.6  Shift Dependent Resources ................................................... 68  
   2.7  Pooled Resources ............................................................... 69  

3  Advanced Models ................................................................................... 73  
   3.1  Visual Logic ...................................................................... 74  
   3.2  Editing Visual Logic ............................................................ 79  
   3.3  Simul8 Profit ..................................................................... 84  

4  Simulation Modeling with Arena ............................................................... 89  
   4.1  Let’s Start Arena .................................................................. 90  
   4.2  Arena’s Building Blocks ...................................................... 95  
   4.3  Building the Model in Arena ................................................ 101  
   4.4  Running the Model ............................................................. 106  
   4.5  Analyzing the Results ......................................................... 110  

5  Intermediate Modeling in Arena ............................................................... 119  
   5.1  Attributes and Assign Modules ............................................. 120  
   5.2  Sub-Models ....................................................................... 123  
   5.3  Scheduled Arrivals ............................................................. 124  
   5.4  Decide Modules ................................................................. 125  
   5.5  Scheduled Resources .......................................................... 126  
   5.6  Sets of Resources ............................................................... 127
# Table of Contents

6  Input Analysis for Simulation Models ................................................................. 130
   6.1  Input Analysis with Data ................................................................. 131
   6.2  Stat::Fit for SIMUL8 ................................................................. 132
   6.3  Arena's Input Analyzer ................................................................. 140
   6.4  Input Analysis without Data ......................................................... 145

7  Making Decisions with Simulations ................................................................. 150
   7.1  Picking the Best Alternative ......................................................... 151
   7.2  Optimizing Your System ............................................................... 160
   7.3  OptQuest for Simul8 ................................................................. 165
   7.4  OptQuest for Arena ................................................................. 174

8  Appendix ........................................................................................................... 177
   8.1  SIMUL8 Professional Services ..................................................... 178
   8.2  Example Models in Simul8 ............................................................. 181
1 Simulation Modeling with Simul8

We will teach you some new tools and techniques that you can use to help in your work. If these tools are not new to you, and to a lot of you they might not be, we will try and help you to understand better when to use them and how to use them.

- This will not be a course in statistical or mathematical theory.
- We will teach you what simulation models can do, how to make build them and how to use these models to make decisions about the systems you work with.
- The software tools we will use are Simul8 and Arena, the two leading system simulation packages on the market
- We will teach through example and by modeling Capital One systems.

This course requires a background in basic statistics and Windows based programs. We can review background material as needed, so please ask if something is confusing you.
1.1 What is simulation?

A simulation is an imitation of a real world system or process. Computers are used to handle the necessary computation allowing us to concentrate on building valid models and analyzing them to get the answers we need.

Capital One has created a Center of Excellence in simulation. The Center aims to provide support that allows the various functional areas within Capital One to leverage simulation as a tool for improving their business processes.
Simulation is a Hammer

Simulation is a hammer.

“SIMUL8 is capable of simulating virtually any process in virtually any amount of detail. Most of the time you need only a fraction of this power to deliver the results you need to make decisions with confidence.”

“The Arena product suite is designed for use throughout an enterprise, from strategic business decisions, such as locating capacity in a supply chain planning initiative, down to operational planning improvements, such as establishing production line operating rates.”

Building a simulation requires an investment of analysts’ time and effort, data support from the “techie” types and managerial support to make sure the project results are actually used. Be sure the investment is worth it.

We will kick off by looking at some very basic models in Simul8, but first let’s examine the steps in building a simulation model?

Step 1 – Decide on the Purpose

Decide on the purpose of the simulation and the performance metrics you want to monitor.

Very rarely will you need to simulate your entire business. Decide what information you need and then determine what part of your operation you should simulate to provide you with that information.

Examples of metrics that you could monitor are:

- Work-in-progress
- Call abandon rates
- Customer wait times
- Equipment utilization
- Completed products count
- Cycle times
- Labor utilization

\[5 \text{ Steps to Simulation by Simul8 Corporation, Herndon, VA, www.simul8.com.}\]
Step 2 - Build a First Pass Simulation

Good data is essential for building an accurate simulation. As the saying goes, “Garbage In, Garbage Out”.

An effective method to make sure that you have the right data for your simulation is to build a first pass model and then collect the data it requires. This ensures that you collect the required data AND means you waste less time collecting nonessential data.
Step 3 - Calibrate Your Simulation

You will need to make sure that your base simulation behaves like the real thing.

Run your simulation and see how the performance measures match the behavior of your existing layout or, if you are simulating a proposed layout, the expected performance. This is where you may need to collect and/or act upon additional data to make sure that the simulation incorporates the variability of your real or anticipated layout.

SIMUL8 lets you adjust your model easily with one click of the mouse. The properties of an object are displayed and changes to the name, timing, routing or other parameters of that object can be made.
Step 4 — Analyze the Results

SIMUL8 lets you answer different questions on your PC before they’re implemented.

Questions like:

- How will changing the batch affect production?
- How will adding an additional operator affect the abandon rate?
- Will a faster machine eliminate our bottleneck?

After the simulations are run you can compare the metrics and implement the best solution based on the reported results.
Step 5 - Share Your Simulations

The final stage to any simulation project is to share the results with managers, colleagues and other involved personnel.

Demonstrating the simulation in action has a good impact and increases understanding and acceptance of any proposed changes. The sophisticated results options and reporting features of SIMUL8 make it easy to share relevant, timely information with your colleagues.
1.2 Let’s Start Simul8

Start Arena. Go the Start button and find the Simul8 program group.

Start > Programs > Simul8 > Simul8

- Graphics Panel
- Model Starter
- Model Workspace
- Standard Windows Menus
- Toolbars
The Standard Windows Menus include the File, Edit and Help menus that are common to all Windows programs. The Clock, Trials, Results, Objects, Graphics and Tools menus are specific to Simul8.

- The Clock menu contains run control commands, e.g. Run, Step and Reset.
- The Trials menu contains options for controlling full length runs of the model for statistics collection.
- The Results menu contains options for statistics collection.
- The Objects menu contains commands and tools for working with the objects in a simulation model.
- The Graphics menu contains options for modifying the appearance of a simulation model.
- The Tools menu gives access to the various wizards, assistants and supporting programs that can be used with Simul8.
Toolbars

The Toolbars include a File/Edit, a Run, a Build, a Simulation Speed and a Graphics toolbar.

- The Standard toolbar contains commands for opening, closing and printing models.
- The Run toolbar contains run control commands.
- The Build toolbar contains the basic objects used to build a simulation model in Simul8.
- The Simulation Speed toolbar gives control over the animation speed of a model.
- The Graphics palette allows you to dress up your model.
1.3 A Simple Model

To start simple, consider processing credit card applications. The applications are scanned at one location and then the image is sent to the applications processing center. The scanners scan the application into the system. There are currently five such scanning machines operating 24 hours per day. Applications come along at intervals of 6 seconds on average, but the system is not consistent and upstream variation makes the arrivals fairly random. A scanner can scan the applications in 25 seconds, plus or minus 5, with the current scanner.

The group manager for credit card applications processing has on his desk a proposal for a new, faster scanner. The new machine will cost $25 per day per machine, instead of $20 per day for the old machine, but will scan in an average time of 20 seconds, plus or minus 5. Is this a good investment?
Elements of a Simulation Model

The basic elements of a simulation model in Simul8 are:

- **Work Entry Points** where the entities or items to be worked on are created in the simulation model. This is a boundary between the system you are modeling and the outside world. E.g. credit card applications arriving from the mail processing area.

- **Work Centers** where the work is performed and value is added to the items in your model. E.g. the scanning centers.

- **Work Exit Points** where the entities or items leave the system you are modeling, another boundary with the outside world. E.g. credit card applications moving on to the next step in processing.

However, we do not have infinite resources with which to perform the work.

- **Resources** are needed to perform the work, but they are scarce or limited. E.g. the scanning machines.

Resources cost money, so we cannot always add resources as needed; this causes delays.

- **Storage Bins** (or queues) are locations where the entities can be held while awaiting the required resources. E.g. the pile of applications waiting for a free scanning machine.

Minimizing delays is a major way companies can increase profits, but there is a trade-off with the cost of the resources.

Simulation is a tool for making these trade-offs.
Mapping Out Our Model

Always start with a pen and paper and think your way through a model before turning to the software. Let’s use a flowchart approach to think our way through the model.

Create  Process  Exit

Storage  Decide

Use the space below to map out your understanding of the scanning system.
1.4 Simul8’s Building Blocks

Now that we have a conceptual model, we need to understand the objects Simul8 uses to build a model.

- Work Entry Points
- Work Centers
- Work Exit Points
- Resources
- Storage Bins

Let’s go through the basic options for each object.
Work Entry Points

The applications will arrive in the model through a Work Entry Point. Work Entry Points need to be told how often to create entities (applications) and how many. They can also be told if there are any labels to attach to the entities, what statistics should be collected about them and where they go next.

- **Name** – the work entry point so that the screen title will change and you can use an identifiable name to refer to it in the model.
- **Input Work Item Type** – is to be left alone for now!!!
- **Inter-arrival times** – are the times between arrivals
  - **Distribution** – specifies the probability distribution that represents the values the inter-arrival times can take and how likely each value is.
  - **Average** – appears when Exponential is chosen for the Distribution. Change the Distribution and the parameters to be filled in will also change.
- **First at start time** – makes the first entity arrive at the beginning of the simulation run if checked. Otherwise a sample will be drawn from the Inter-arrival time Distribution.
- **Unlimited arrivals** – makes entities arrive as soon as the downstream objects can handle them.
We will model each scanning machine using a Work Center. It will delay the applications for the required amount of time and only take one at a time. Work Centers need to be told how long the work will take for each entity arriving to be worked on. Options are available for handling arrivals from multiple objects in the model, sending finished work on, what resources are needed for work to be performed and what statistics to collect about the work. Work Centers can also be told how often they break down and how long repairs take.

- **Name** – the work center so that the screen title will change and you can use an identifiable name to refer to it in the model.
- **Timing** – specifies the time taken to perform the work on one entity.
  - **Distribution** – specifies the probability distribution that represents the values the work times can take and how likely each value is.
  - **Average** – appears when Exponential is chosen for the Distribution. Change the Distribution and the parameters to be filled in will also change.
- **High Volume** – makes Simul8 approximate the operation of the model when very large numbers of entities will pass through the model. If not checked, each entity is handled individually.
Simulation models need somewhere to dispose of finished work. The applications will finish up at the Work Exit Point, allowing Simul8 to calculate statistics, such as the total time in the system. Work Exit Points work just fine if left alone. They can be told what results to collect and how they should appear in the model.

- **Halt Model at Limit** – makes Simul8 end the simulation trial once a specified number of entities reach them.
- **Segregate Results** – specifies that statistics should be broken down by defined criteria.
- **High Volume** – makes Simul8 approximate the operation of the model when very large numbers of entities will pass through the model. If not checked each entity is handled individually.
If our scanning machine needed an operator to work it, we could model the machine itself as a Work Center and tell that Work Center that it requires an operator Resource to be able to perform work. This would only affect the statistical results if there were fewer operators than scanning machines or if the operators took breaks for instance. In this model, we do not need to use a resource, but they will come in very useful in more complex models.

Resources need to be told what each resource is called and how much of each there is. They can also be told how many of each type of resource is available, whether the availability of resources changes over time according a schedule and whether they belong to a set of pooled resources.

- **Name** – the resource so you can use an identifiable name to refer to it in the model.
- **Shift Dependent** – means that the number of resources available depends on a schedule. Shifts can be named and their duration and the number of resources available during that shift specified.
- **Pool Resource** – specifies that resource is part of a pool or set of resources, any of which can be used by a Work Center.
Storage Bins

A Storage Bin is the place where the applications wait till the scanning machine is ready for the next application. They are a queue. Storage Bins need to be told how the queue works. They can also be told how many items are stored at startup and what results should be collected.

- **Name** – the Storage Bin so that the screen title will change and you can use an identifiable name to refer to it in the model.
- **Capacity** – is the maximum number of entities that can wait here. This can be infinite.
- **Shelf Life** – is the maximum time that entities can be stored before they must be sent to a Work Center that is told to expect Expired Only entities.
- **Min Wait Time** – specifies the least time that an entity can spend in the queue.
- **Prioritize** – specifies that entities should leave the Storage Bin according to a specified order of some label attached to them.
- **LIFO** – means the last entity to arrive is the first to leave (Elevators).
- **High Volume** – makes Simul8 approximate the operation of the model when very large numbers of entities will pass through the model. If not checked each entity is handled individually.
- **Segregate Results** – specifies that statistics should be broken down by defined given criteria.
1.5 Building the Model in Simul8

Let’s build the simplest model.

Step 1 – Drag and drop a Work Entry Point onto the Simulation Window.

Step 2 – Drag and drop a Storage Bin onto the Simulation Window.

Step 3 – Drag and drop a Work Center onto the Simulation Window.

Step 4 – Drag and drop a Work Exit Point onto the Simulation Window.

Notice that the objects are automatically connected with Route arrows to show the flow of entities. If the automatic arrows are wrong, click on the arrow and press the Delete key.

Select a Route arrow and press the Delete key.

Select the Route Drawing Mode.

Click on the first object you want to connect with the pointer.

Click on the second object you want to connect with the pointer.

Unselect the Route Drawing Mode by clicking on it again.

Repeating an existing route connection deletes the route arrow.
Specifying the Work Entry Point

- Double click on the Work Entry Point

- Name the Work Entry Point Arriving Applications

- For the Inter-arrival times, choose Exponential for the Distribution

- Change the Average to 6

- Click OK

Remember this is 6 seconds. To change the time units to seconds:

- Exit the Work Entry Point Properties dialog.

- Click on the clock.

- Choose Seconds as the Time Units.

To make the title appear in the Simulation Window:

- Click on the Graphics button.

- Click on the Title button.

- Check Show Title on Simulation Window.

Are there any other options you think we should be using?
Specifying the Work Center

- Double click on the Work Center

- Name the Work Center *Scanning Machine*

- For the Timing, choose Triangular for the Distribution

- Change the Lower to 20, the Mode to 25 and the Upper to 30

- Click OK

Show the title on the screen.

Are there any other options you think we should be using?
Specifying the Storage Bin

Double click on the Storage Bin

Leave the Name of the Storage Bin as Queue for Scanning Machine

This is the default name now that you have named the upstream Work Center that causes the delay of items in this Storage Bin. Show the title on the screen.

Are there any other options you think we should be using?
Specifying the Work Exit Point

- Double click on the Work Exit Point

- Name the Work Exit Point: Scanned Applications

- Click OK

Show the title on the screen.

Are there any other options you think we should be using?
1.6 Running the Model

Your model should now look like this. You may need to move the objects around so you can read the titles.

Your model should now look like this. You may need to move the objects around so you can read the titles.

Click on Run

You will probably need to slow the animation down by sliding bar on the Simulation Speed toolbar.

Is anything going wrong?

Stop the simulation by clicking on Run again.

Select the Storage Bin.

Click on the Time Graph button.

Why is the queue exploding?
Representing Multiple Work Centers

We need to represent the five scanning machines.

- Double click on the Work Center
- Click on Replicate = 1
- Change the Effective number of work centers to 5

This means that we have five identical Work Centers working that can scan applications.

- Click on Run

This is what the graph should look like after a short while.
Running the Model Again

In Simul8, the Run command is like a play button on a DVD player. When pressed the model runs. However, if it is pressed again it just pauses the simulation even though the hint for the button now says Stop. As you can see from the graph of the number in the queue, it does not reset the simulation.

To reset the simulation to the beginning, click on Reset Clock to Start 🔄. The counts will be set to zero and the time graphs will be wiped clean.
1.7 Analyzing the Results

Once the simulation has run for a while, the Time Graph of the number in the Queue for Scanning Machine should look something like this.

The graph shows the change in the number of people in the queue over time. Notice that the graph jumps in value at specific times. A jump up is an arrival of an application at the Work Entry Point. A jump down is an application that has finished scanning at the Work Center, thus allowing the next application in the queue to be brought to the scanning machine.

The dashed, horizontal line is the average number of people in the queue so far. It is a time average, so it is calculate by finding the area under the graph and dividing by the total run time.

If you had to report to your manager the average number of applications being delayed at any given time, would this value be a good estimate?
Warming Up

Notice that at the beginning of the simulation run, there are no applications in the queue. The queue starts empty and the first five applications to arrive can go straight to one of the five scanning machines.

The actual system works twenty-four hours a day, so this initial period in the simulation is not representative of anything that actually occurs in the system. The simulation is just warming up.

When collecting statistics, this empty period biases the average number in the queue statistic down. We should not collect data during this period. This is called the Warm-Up Period.
Specifying the Warm-Up Period

To specify a warm up period:

 diffé Choose Clock > Warm-Up Period.

You will get the Warm-Up Period dialog

 diffé Type 100

How did I decide to use a value of 100?
Determining the Warm-Up Period

If we specify the Results Collection Period to be 500 (Clock > Results Collection Period) and re-run the simulation with different random values ( ), then we will get different time graphs. Eyeballing the graphs, we will be pretty safe if we don’t collect statistics for the first 100 seconds.

Warming Up

Simul8 specifies the Warm-Up Period and the Results Collection Period as disjoint times.
**A Complete Run**

With a Warm-Up Period of 100 and a Results Collection Period of 500, allow the simulation to run to the end. The simulation will stop itself and show you the counts recorded in the 500 seconds of the Results Collection Period.

- 🎨 Double-click on the Work Entry Point.
- 🎨 Click on Results.

You will get a window with the results of the statistics collected for this object.

<table>
<thead>
<tr>
<th>Work Entry Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arriving Applications</td>
</tr>
<tr>
<td>Number of work items entered</td>
</tr>
<tr>
<td>Number of work items lost</td>
</tr>
</tbody>
</table>

This tells us that 71 work items (entities) arrived during the Results Collection Period and none were lost.

How would Simul8 lose work items?

The Results button is available in each object’s properties dialog. Let’s look through the others.
Storage Bin Results

The Storage Bin Results look like this:

There are currently no work items (entities) in the storage bin. During the run there were a minimum of 0 (for instance at the beginning) and at most 4 items in the storage bin. On average there were 0.22 items in the bin. 71 work items entered the storage bin.

Queuing Time statistics are also available. Items were delayed in the storage bin for between 0 and 13.98 seconds. Looking at the distribution of the delay times, they had an average of 1.45 and a standard deviation of 3.13. For the 19 items that actually had to wait in the storage bin (non-zero queuing times), the shortest delay was 0.31 and the average was 5.41. Obviously the maximum non-zero queuing time is still 13.98.
Storage Bin Results

Graphs of the statistic over time are available using the button, while histograms of the collected data can be obtained using the button. We have already seen a time graph of the number of work items in the queue.

Let’s look at the histogram of the queuing times.

This is the distribution used to calculate the average and the standard deviation of the queuing time.
Work Center Results

The Work Center Results look like this:

There are currently 4 applications being scanned. The number being scanned varied from 0 (at the beginning) to 5 (there are five scanning machines). On average there were 3.48 applications being scanned. Of the 71 applications that arrived in the system, 67 completed scanning.

Look at a time graph for the number of work items in the scanning machine Work Center:

It looks like the five scanning machines are all busy for some periods of the simulation run.
Work Complete Results

The Work Complete Results look like this:

66 applications left the system. For these 66 completed applications, the shortest time through the system was 22.94 seconds, the longest 39.88 seconds. Looking at the histogram of the Time in system:

The average time in the system was 28.35 seconds and the standard deviation was 3.54 seconds.
**Missing Work Item**

Hang on!!! 71 applications arrived in the system; 67 finished scanning; 4 are still at the scanning machines; 66 left the system. What happened to the other one?

The other work item is still in transit. Do the transits take time?

The average queuing time was 1.45 seconds; the average scanning time was set to be 25 seconds; the average time in the system was 28.35 seconds. Where did the extra 1.89 seconds come from?

Choose Graphics > Distance / Travel time.

You will get this dialog.

Simul8 assumes that the length of the routes means something. 10 pixels equal 1 foot. The entities travel at 10 feet per simulated second. You can modify these defaults to suit your model if you wish. You can change the units of measurement in the File > Preferences dialog using the Distance tab.
Modeling Exercise

- Select a Route arrow and press the **Delete** key.
- Select the Route Drawing Mode 🌓.
- Click on the first object you want to connect with → the pointer.
- Click on the second object you want to connect with → the pointer.
- Unselect the Route Drawing Mode 🌓 by clicking on it again.

Notice the numbers that pop up briefly. The L: number is the length of the route (in feet if you have not change this default preference). The T: number is the time an entity will take on this route (in seconds as we have change the Time Units in the Clock dialog).

- Select a Route arrow.
- Double-click on the selected Route arrow.

Notice that a black square has appeared on the route arrow where you clicked on it. You can use this square to drag the route into different shapes. This, of course, changes the length of the route and the transit time for entities on that route.
Results Summary

- Click on Results Summary.

You will get the default summary of the results.

The four objects are listed. Double-clicking on one brings up the Results dialog that we have already seen. Within this dialog if you right-click on any number, that statistic will be added to the Results Summary. This was actually possible when we opened the Results dialogs from the objects’ properties dialogs too.

- Click on Add all results to Results Summary Window.
Full Results

The Results Summary will now have all possible statistics for all objects in the model.

How good are these estimates? This is simulation, so they can’t be exact. What level of confidence can we have in them?

Click on Multiple Runs (a Trial).
The 95% confidence intervals, Low and High are added to the Results Summary.

<table>
<thead>
<tr>
<th>Research Results Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scanned Applications</td>
</tr>
<tr>
<td>Average Time in System</td>
</tr>
<tr>
<td>Scanned Applications</td>
</tr>
<tr>
<td>Number Completions</td>
</tr>
<tr>
<td>Scanned Applications</td>
</tr>
<tr>
<td>“In System less than”</td>
</tr>
<tr>
<td>% In System less than</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>Working %</td>
</tr>
<tr>
<td>Blocked %</td>
</tr>
<tr>
<td>Stopped %</td>
</tr>
<tr>
<td>Number Completions</td>
</tr>
<tr>
<td>Minimum Utility</td>
</tr>
<tr>
<td>Average util</td>
</tr>
<tr>
<td>Maximum util</td>
</tr>
<tr>
<td>Current Contents</td>
</tr>
<tr>
<td>Change over %</td>
</tr>
</tbody>
</table>

How were these confidence intervals obtained?
**Trials**

We have set it up so that each run of the simulation model has 100 seconds of warm-up and 500 seconds of results collection. At the end of each run we get average statistics for time in system, number in queue, … To obtain confidence intervals, we would have to perform multiple runs to find out what the variability is in these average statistics across different runs. Simul8 calls a set of runs a Trial.

_choose Trials > Conduct Trial_

You will get the Conduct Trial dialog.

_save your model as ScanningMachines1.s8_
Analysis Exercise

What effect does the number of runs have?

- Change the Number of runs in trial to 25.
- Click on Multiple Runs (a Trial).

What happens to your confidence intervals?

- Change the Number of runs in trial to 100.
- Click on Multiple Runs (a Trial).

What happens to your confidence intervals?

Continually increasing the number of runs would give you better and better estimates of what? Is this the estimate we want?

- Change the Number of runs in trial to 5.
- Change the Results Collection Period to 5000.
- Click on Multiple Runs (a Trial).

What is this estimate of?
Modeling Exercise

We have five scanning machines. We have chosen to represent this by making five replicates of the work center.

- Change the number of replicates of the Work Center back to 1.

There is another way to do this. Actually model five separate Work Centers. Here is a quick way to do this.

- Select the Scanning Machine Work Center.
- Hold down the CTRL key.
- Drag the Work Center and drop it below.
- Repeat to get five Scanning Machines.

If you have the routes showing, you will see that the entities have five routes out of the Storage Bin to the different Work Centers and then converge back to the same Work Exit Point. The Storage Bin will automatically send the work items to the next available Work Center.
Modeling Exercise

You can also drag a box around multiple objects, hold down the `CTRL` key and drag and drop copies of the selected objects.

- Delete the copies of the Scanning Machine Work Center.
- Select the Work Center and the Storage Bin.
- Hold down the `CTRL` key.
- Drag either object and drop it below.
- Repeat three more times.

You will now have five separate Work Centers, each with their own queue.

- Click on Run.

What is going wrong?

- Double-click on the Work Entry Point.
- Click on Routing Out.

The current selection is Priority. This takes the top destination in the list. As the Storage Bin never gets full, it keeps taking new arrivals.

- Choose Shortest Queue.
- Click on Run.

Now it’s working.

Which model is more representative of the actual system?
Modeling Exercise

We are creating this model with the objective of determining whether this new faster scanner is worth the investment. The new scanner can scan documents in an average of 20 seconds plus or minus 5.

We have the system with the old scanner modeled. Model the system with the new scanners.

When you are done:

☐ Save your new model as ScanningMachines2.s8
**Analysis Exercise**

Compare the results.

- Which system would you recommend?
- Is there any further information you need to make a final decision?
- How can we obtain this information?
Further Exercises

Your manager likes your recommendation, but has just found out some new information. The new scanners are less reliable.

- The old scanners would break down every 24 hours of operation on average.
- The new scanners will break down every 18 hours of operation on average.
- Each scanner takes an average of 30 minutes to fix.

For each Work Center representing a scanner in the original model,

- Open the properties dialog.
- Click on Efficiency.
- Choose the Detailed radio button.
- Click on Time between Breakdowns.
- Keep the Distribution as Exponential and type 86400 for the Average.
- Click on Time to Repair.
- Keep the Distribution as Exponential and type 1800 for the Average.

Repeat these modifications for the new scanner model, changing the average Time between Breakdowns to 64800.

Which system would you pick now? Can you get the answer running the simulation model for the same Results Collection Period?

---

MTBF stands for Mean Time Between Failures. MCBF stands for Mean Cycles Between Failures.
Analysis Exercise

In queuing analysis, if we have \( \lambda \) arrivals to the system per time unit and \( s \) servers who can each serve \( \mu \) entities per time unit, then we can find various measures for a steady-state (or constantly running) system.

First calculate the probability that there is no one in the system at any given time, denoted by

\[
\pi_0 = \frac{1}{\sum_{j=0}^{s-1} \frac{(s\rho)^j}{j!} + \frac{(s\rho)^s}{s!(1 - \rho)}}
\]

Then calculate the probability that all the servers are busy, denoted by

\[
P(j \geq s) = \frac{(s\rho)^s}{s!(1 - \rho)} \pi_0
\]

- The total time an entity spends in the system is \( \frac{P(j \geq s)}{s\mu - \lambda} + \frac{1}{\mu} \).
- The time an entity spends in the queue \( \rho P(j \geq s) \).
- The number of entities in the system \( \frac{\rho P(j \geq s)}{1 - \rho} + \frac{\lambda}{\mu} \).
- The number of entities in the queue \( \frac{P(j \geq s)}{s\mu - \lambda} \).

Luckily for you we have done this part for you. Take a look at the spreadsheet MMs.xls.

What would these measures be for the scanning system? Do you know what assumptions are being made to get these measures?
We will build a much more advanced model, a Call Center model. There will be multiple types of resources, some working on schedules and some pooled based on their capabilities. There will be many parts to this model, so the logic of the model will be broken down into sub-models.

First we will introduce these new concepts, then we will build the whole model as an exercise.
2.1 The Call Center

The call center supports three primary functions: supplying basic account information, responding to credit increase requests and supplying technical support for online account services. There are 26 trunk lines to take calls in to the call center; if these are busy then the caller gets a busy signal. The caller first hears a recording and is asked to press one for account information, two for credit increase requests and three for technical support. About 76% of customers press one, 16% press two and 8% press three. The time for this activity is uniformly distributed between 0.1 and 0.6 minutes.
Account Information

Customers requesting account information are primarily handled by an automated system. The time (in minutes) for this operation follows a triangular distribution with parameters 2, 3, and 4. 10% of the callers opt out of the automated system in favor of speaking to a person and 15% of callers who use the automated system still want to speak to someone afterwards. These conversations both last for a time following a triangular distribution with parameters 3, 5, and 10 (minutes). All customers then exit the system.
Credit Increase Requests

The account information personnel handle credit increase requests. If no one is available, the customer is treated to some soothing, new age music. The call duration is different for customers who are accepted and those who are rejected. The call durations for rejected customers follow a triangular distribution with parameters 4, 15 and 45 (minutes). This consists mainly of complaining. Accepted customers calls follow a triangular distribution with parameters 4, 8 and 12 (minutes). All customers then exit the system.
Technical Support

Technical support is the complex part of the model. There are three types of product with customers asking for *product one* 25% of the time, *product two* 34% of the time and *product three* 41% of the time. An automated message asks for the customers’ choice, a process that takes a uniformly distributed amount of time between 0.1 and 0.5 minutes. The technical support staff is not all trained on all products, so there are certain sub-groups that can handle each product. If no one is available for that product, then the customer joins an electronic queue and is subjected to loud rock music chosen by the head of the technical support staff.

The calls themselves last for a time that follows a triangular distribution with parameters 3, 6 and 18 minutes. However, 5% of the calls require additional work. For these calls, the support case description is sent to a specialized group who work on the problem for an average of 60 minutes (exponentially distributed). The case is then transferred back to the same technical support person who handled the initial call, who calls back the customer taking an additional time following a triangular distribution with parameters 2, 4 and 9 minutes. Callbacks take priority over new calls.
### Callers Per Hour

The call center takes calls between 8:00 am and 6:00 pm. The call arrival rate is not homogeneous over the day. The following table contains the average call rates per hour over the day:

<table>
<thead>
<tr>
<th>Time</th>
<th>Calls per Hour</th>
</tr>
</thead>
<tbody>
<tr>
<td>8:00 to 10:00</td>
<td>45</td>
</tr>
<tr>
<td>10:00 to 12:00</td>
<td>75</td>
</tr>
<tr>
<td>12:00 to 2:00</td>
<td>100</td>
</tr>
<tr>
<td>2:00 to 4:00</td>
<td>85</td>
</tr>
<tr>
<td>4:00 to 6:00</td>
<td>55</td>
</tr>
</tbody>
</table>
Technical Support Schedules

All technical support personnel work an eight-hour day with a half-hour lunch break. There are 11 personnel in all. They work in three staggered shifts, early morning, mid-morning and late morning.

- The early morning shift works from 7:00 am to 11:00 am and then takes a half-hour break. They then work from 11:30 am to 3:00 pm.
- The mid morning shift works from 9:00 am to 1:00 pm and then takes a half-hour break. They then work from 1:30 pm to 5:00 pm.
- The late morning shift works from 11:00 am to 3:00 pm and then takes a half-hour break. They then work from 3:30 pm to 7:00 pm.

In each shift there is one person who is trained for each product type. There is also a manager who is trained in all three product-types on the early morning shift and another, similarly trained, on the late morning shift.
New Modeling Issues

We have a number of new modeling issues that need to be handled in this model:

- **Labels** – We need to be able to differentiate between the call types and product types, so the entities need to be labeled.

- **Sub-windows** – This model is complex enough that it will be worth breaking the model up into pieces. The overview could be shown to managers and the detail kept for the analysts.

- **Time Dependent Arrivals** – The time between arrivals does not stay constant over a day.

- **Routing Out** – The entities will not follow simple paths through the model. We will have to control their destinations when leaving each part of the model.

- **Shift Dependent Resources** – The availability of resources will vary over the day.

- **Pooled Resources** – The resources capable of handling the work depend on training. We will need groups of resources that can be called upon for each product type.
2.2 Labels

Labels are a useful modeling tool in Simul8. They are attributes of the entities in a model that are passed along with the entities and can be accessed by the objects in your model.

**Controlling Work Flow:** In this case we will have a label that identifies the type of call and a label that identifies the product type. These labels will be used to control the flow of work. Labels can be used to filter entities that can enter a Work Center in the Routing In dialog. Labels can be used to specify where an entity goes after leaving an object (Routing Out).

**Entity Specific Processing Times:** We will also have labels that identify whether the credit increase requests were accepted or rejected and labels that contain the parameters of the corresponding call length distribution. These will be used to determine the processing time at a Work Center. Labels can be used in place of numbers in many dialogs, especially the parameters of distributions.

**Segregating Results:** The product type label can be used to segregate the results of the Storage Bins and Work Centers in the model. Thus we can analyze the model’s results separately for each product.

**Prioritized Queues:** Simul8 has a system label called lbl Importance that can be used to specify priorities in a Storage Bin. We can use this for the callbacks that take priority over the new calls.

**Entity Specific Shelf Life:** Another feature that we could use here is random Shelf Life. Customers waiting in the queue for technical support could hang up after a given amount of time. This is called Shelf Life and is specified in the Storage Bin. However, we could assign a random Shelf Life to a label and then use the label name in the Storage Bin dialog.
Creating Labels

To create a label:

- Choose Objects > Labels.

You will see the List of All Labels in this simulation.

- Choose New.
- Type Call Type for the name.
- Choose Number.
- Press OK.

To specify values for a label each object’s properties dialog has a Label Actions button. For instance, when entities are created in a Work Entry Point, we can specify the call type.

- Press Label Actions.

You will see the Actions dialog.

- Press Add.
- Choose Call Type and press OK.

You can point to one of the selections for Label Action and a hint will appear describing its use.

- Select Set to.

We now need to specify the distribution.
Specifying Distributions

The caller first hears a recording and is asked to press one for account information, two for credit increase requests and three for technical support. We will set the Call Type to 1 for account information, 2 for credit increase requests and 3 for technical support. About 76% of customers request account information, 16% request a credit increase and 8% request technical support.

To create this distribution we actually use Simul8’s Distribution Wizard. Luckily, we can get to this by:

- Press New under Distribution.

Again, point to one of the selections for the type of the distribution and a hint will appear describing its use.

- Select Probability Profile and press Next.
- Type Call Type Distribution for the name of the distribution.
- Keep the default Discrete distribution.
- Click on a bar of the probability plot and drag it up and down.

You will see the Value change when you click on a bar. As you drag it up and down the Percent will change. Make sure you have the right percentages for 1, 2 and 3 and that the other values are set to 0%. Once you are done, you can press OK to get out of the dialogs.

You can view all named distributions in a simulation model by going to Objects > Distributions.
2.3 Sub-Windows

To break up a complex model, sub-windows can be used to separate the pieces. For instance, we could have the following sub-windows:

- Call Arrivals;
- Account Information;
- Credit Increase Requests;
- Technical Support.

To create a sub-window,

- Drag a box around the set of model objects you want in the sub-window.
- Right-click and choose Create Sub-Window.

You can drag and drop objects into and out of a sub-window.

To name the window and set the graphics:

- Right-click in the window area and choose Window Properties.
- Type, for instance, Call Arrivals.
- Under Image, press Select.
- Press New.
- Type, for instance, Call Arrivals Image.
- Press Library.

You can choose many types of image. I chose ANSWERING.bmp. If you OK out of the Window Properties, and close the sub-window, you will see the image is used to represent the sub-window. Double-click on the image to open the sub-window.
2.4 Time Dependent Arrivals

Arrival rates that vary over time are called Time Dependent in Simul8. They are set up using the Distribution Wizard. Create a new Work Entry Point and for the Distribution, press New.

Choose Time Dependent and press Next.

You will see the Time Dependent Distribution dialog.

Press Add.

Keep the From Time as 8:00.

For the distribution, press New.

Type Arrivals at 8 for the distribution’s name.

Choose Exponential for the Distribution.

Type 1.3333 for the Average.

The average is the time between arrivals, so if we have 45 calls per hour on average, then we have 60/45 = 1.3333 minutes between calls on average.

Go ahead and complete the rest of the arrival rates.
2.5 Routing Out

We have already seen one option for Routing Out, namely the shortest queue. Here we will direct entities based on their label values. We want to send the different call types to different parts of our simulation model. To demonstrate this method, add three Storage Bins to represent where the different call types will go next.

- Add route connections from the Work Entry Point to each Storage Bin.
- Double-click on the Work Entry Point.
- Click on Routing Out.
- For the Discipline, choose Label.

The List of All Labels in this simulation will appear.

- Choose Call Type.
- Press OK.

Simul8, will now send an entity with a Call Type label equal to 1 to the first destination in the To list. An entity with a Call Type label equal to 2 to the second destination in the To list, etc. You can change the order of the To list by selecting a destination and using the up and down pointing hands.

Again, point to the selections for a discipline and a hint will appear describing its use.
2.6 Shift Dependent Resources

The technical support personnel work according to shifts. Thus the availability of resources will change over the day.

- Add a new resource to the model.
- Open the resource properties dialog and name the resource EM 1.
- Check Shift Dependent.

We can use the Shift Availability to choose shifts and specify the availability of this type of resource during this shift.

- Press Shift Work Patterns.
- Press New.
- Enter Early Morning for the shift name, 07:00 for the Start Time and 15:00 for the End Time.

What happened to their half-hour break? What could we do to give back the break? Is the additional modeling worth the effort?
2.7 Pooled Resources

Go ahead and create 11 resources. There are four on the early morning shift. I called them EM 1, EM 2, EM 3 and EM M. EM 1 stands for early morning person trained on product 1. EM M stands for the early morning manager\(^5\). Remember there is a manager on the early morning and the late morning shifts and they are trained for all three products.

The work center for *product one* technical support calls is going to need to seize a person trained for *product one*. If it cannot, then the customer will have to wait in the electronic queue and listen to annoying music. However, the work center can seize any of the *product one* trained personnel who might be working at that time. It can choose from a pool of people.

- Add a new resource to the model.
- Open the resource properties dialog and name the resource *Product 1*
- Check Pool Resource.
- Move all the product-one trained resources from the available resources to the Resources in this Pool.

Choose the pooled resource in the Work Center for product one and work can only proceed once one of these five resources is available.

\(^5\) An easy way to copy objects in Simul8 is the Duplication Wizard. Right-click on the object and choose Duplication Wizard. Choose the number of copies and the offset to place each one.
Modeling Exercise

Complete the Call Center simulation model in Simul8. Verify that your model meets all the specifications in Section 2.1.

☐ Save your model as CallCenter.s8
Analysis Exercise

What statistics will be of interest in the Call Center model?

- Determine the statistics of interest.
- Analyze the model’s results.
  - In your analysis do you have to determine a warm-up time?
  - Remember to consider your level of confidence in the results.
- Determine what parts of the system you can control to make it more efficient.
  - Come up with recommendations for improving the real system — to make it more efficient.
- Are there any parts of the model that could be made more accurate or realistic?
  - Come up with recommendations for improving the simulation model — to make it more realistic.
Further Exercises

**Exercise 1:** Modify your simulation to incorporate the half-hour break in each shift. How does this affect the results of the simulation model?

**Exercise 2:** Modify the model so that 50% of all customers who have their credit increase requests rejected ask to speak to a manager. How does this affect the results of the simulation model?

**Exercise 3:** In what order are the pooled resources being taken. Do the managers get taken first or last? Modify the model so that managers are only used if no regular employees are available. Does this improve the problems with disgruntled customers?

**Exercise 4:** Add an additional manager in the mid-morning shift. Does this improve the problems with disgruntled customers?
Special purpose simulation packages, such as Simul8 and Arena, are designed with features that simplify the construction of a simulation model. Writing simulation code in general purpose programming languages, such as C++ and Java, is much more time consuming and complex as you have to tell them how to handle basic simulation events. Despite the specialized functionality of a special purpose simulation package, the simulation modeler always runs into one big problem, the real world. There is always a characteristic of a system that requires functionality beyond the basic objects or modules in the package.

Luckily, Simul8 and Arena acknowledge this problem and give you a solution. They are fully object-oriented programs and so you can access their objects from various programming languages. Microsoft™ designed Visual Basic for Applications (VBA™) for this purpose. With VBA you can control the simulation, accessing the properties of the model and changing them. If you are a decent programmer this is well within your reach. It is, however, a bit beyond this course.

Simul8 offers an even simpler alternative: Visual Logic. Visual Logic is built into Simul8 and Simul8 offers many points of access to this flexible, yet simple, language. In this section, we will give a simple example of one use of Visual Logic.

Simul8 also has a variety of companion packages that have been written to integrate with Simul8 through its objects. One example is Simul8 Profit and we will demonstrate how this useful add-in can be used to add the economics of the system into the simulation model.
3.1 Visual Logic

In examining the simple scanning machine model, the manager of the department realized that he had oversimplified the length of time each scan takes. Rather than a single distribution, there are three different types of applications, taking different periods of time to scan. He is not sure whether it is worth gathering the additional data to model this mixture of applications. As a demonstration, he wants you to model the following mixture. There are equal numbers of each type of application. The first type averages 20 seconds per scan, the second type averages 25 seconds per scan and the third type averages 30 seconds per time. Each type’s scan time can vary by plus or minus 5 seconds.

Does this change make much difference to the model?
Adding Labels

We will start by adding labels for the type of application and the service time. Why do we have a service time label?

We will assign the service time label when the entity is created, just after we determine each entity’s application type label. The entity will continue through the model with a predetermined service time. In the Work Center, instead of specifying a distribution for the Timing, we will used the fixed value specified in the service time label.

1️⃣ Select Object > Labels

2️⃣ Press New

3️⃣ Type Application Type for the label name

4️⃣ Press OK

5️⃣ Press New

6️⃣ Type Service Time for the label name

7️⃣ Press OK twice

You now have two labels attached to each entity. One ready to have the type of application assigned as 1, 2 or 3 and one ready to have the service time assigned from a random distribution that depends on the value of the application type label.
**Application Type Distribution**

For this simple operation, we do not need Visual Logic. First we will create an application type distribution and then we will set the label to take values from this distribution.

- Select **Object > Distributions**
- Press **New**
- Type **Application Type Distribution** for the distribution name
- Select **Probability Profile** for the type of distribution
- Press **Next**
- Set the Value 1 to a Percent of 33.333
- Set the Value 2 to a Percent of 33.333
- Set the Value 3 to a Percent of 33.334
- Make sure the other Values have a Percent of 0
- Press **OK** twice
Assigning Application Types

Now we need to assign the Application Type label to take values from this distribution.

- Open the Arriving Application Work Entry Point’s properties dialog
- Press Label Actions
- Select Application Type in the List of labels
- Select Set To under the Label Action
- Under Distribution, select Application Type Distribution
- Press OK three times
Type Dependent Service Times

Now we need to assign the service time label from the appropriate distribution.

Create three new distributions, called Application 1 Service Times, Application 2 Service Times and Application 3 Service Times.

Set the distributions to Triangular with the appropriate parameters.

Go back to the Label Actions dialog in Arriving Applications.

Select Service Time in the List of labels.

Press Visual Logic.

You will get the following dialog:

Press the Insert Key or right-click in the dialog.

The menu gives you the basic Visual Logic commands.

We will use an IF, ELSEIF, ELSE structure to determine the type of application and SET ... = ... to set the Service Time label to the appropriate distribution.
3.2 Editing Visual Logic

Let’s start with the IF, ELSEIF, ELSE structure.

Select IF ...(ELSE / WHILE etc)

You will get the Conditional Block Editor.

![Conditional Block Editor]

To the right of the first clause, press ➔.

You will get the Formula Editor.

Select Object

Scroll down and double-click on Application Type

![Formula Editor]

Press OK

Leave the condition in the middle of the Conditional Block Editor as =

Type 1 for the last clause

You should now have the following text in your editor

![Edited Condition]

© 18 June, 2004 by Jason R. W. Merrick, All Rights Reserved.
Editing Visual Logic

To complete the IF structure:

- Select the IF statement
- Right-click and select IF ...(ELSE / WHILE etc) again
- In the Conditional Block Editor, select ELSEIF
- Set the clauses to Application Type = 2
- Select the ELSEIF statement
- Right-click and select IF ...(ELSE / WHILE etc) again
- In the Conditional Block Editor, select ELSE

You should now have the following Visual Logic code.

```
 IF Application Type = 1
   Insert logic here to use "IF" condition is true.
 ELSE IF Application Type = 2
   Insert logic here to use if this is FIRST true condition in this "IF block".
 ELSE
   Insert logic here to use if NONE of the conditions in this "IF block" are true.
```
Editing Visual Logic

To complete the code, we must use the appropriate Set statements.

- Select the first Insert Logic statement
- Right-click and select SET ... = ...
- In the Set Value Editor, to the right of Information, press ...
- Select Object and scroll down to double-click on Service Time
- Press OK
- To the right of Calculation, press ...
- Select Object and scroll down to double-click on Application 1 Service Times
- Press OK twice

Change the next two clauses to the appropriate statements
The Complete Code

The complete code should look like this:

At run time, an application will first be assigned an application type and then go through this code to be assigned a random draw from the appropriate service time distribution.

Be sure that the Application Type label is assigned first then the Service Time label. In the Label Actions dialog, you can change the assignment order using the up and down hand arrows.

Save your model as ScanningMachines1VL.s8
Programming Exercises

Exercise 1. Add the modification to the model with the new scanners. Compare the two systems with this mixture of application types. Is it worth the manager’s effort to gather the data necessary to improve the model?

Save your model as ScanningMachines2VL.s8

Exercise 2. When the technical support personnel call back customers, there is a chance that the automated call back system will have the wrong number. Call center analysts have found that a good model for this is a hierarchical probability model. Each call back has a probability of being a wrong number that is drawn from a Beta distribution. This is determined first and then a Uniform random number between 0 and 1 is drawn. If the Uniform random number is less than that callback’s probability of being a wrong number, then the customer is not reached. Otherwise the call goes through.

Add labels, distributions and the appropriate Visual Logic statements to model this scenario in the Call Center simulation.

Save your model as CallCenterVL.s8
3.3 Simul8 Profit

SIMUL8 Profit lets you allocate costs and revenues to objects and activities in the simulation model. As the model runs, SIMUL8 uses all this information to accumulate data on the flows of money generated by the activities taking place and the resources being used. This accumulated information is then displayed in summary form in the Income Statement that can be seen by clicking the $ button on the SIMUL8 tool bar.

The old scanner costs $20 per day, the new scanner $25. Each scanned application is estimated to be worth 10¢ in revenue. Let’s make our comparison using Simul8 Profit. If you have a version of Simul8 with the Profit add-in, you will get an extra menu called Finance and each of the objects’ properties dialogs will have a Finance button.

Here’s how we specify the costs for the scanning model.
Adding Work Center Costs

Enter the actual cost in dollars to buy the Work Center and the cost of using it.

- Open the model in ScanningMachines1VL.s8
- Open the Scanning Machine Work Center’s properties dialog
- Press Finance
- Enter 20 for the Capital Cost and press OK twice
- Open the Scanned Applications Work Exit Point’s properties dialog
- Press Finance
- Enter 0.01 for the Revenue (per Unit) and press OK twice

The text boxes for revenues are shown in green to indicate that this is money earned, not spent.

We have entered the $20 per day for the scanning machine as a Capital Cost, so we need to make the simulation run for one day (or 86400 seconds).

- Change the Results Collection Period to 86400
Examining the Income Statement

With the costs and revenues entered, we can take a look at the results.

- Do a full Trial (multiple runs)
- Select Finance > Income Statement

You can double click on any of the bold categories (Costs, Revenue or Profit) to expand the breakdown. Right now, we are making a daily profit with the old scanner.
Examining the Income Statement

To get confidence intervals on the cost statistics:

- Right-click on the three categories

Each time you do this you will get a message telling you that the Income Statement category has been added to the Results Summary.

We are 95% confident that the average daily profit is between $41.02 and $44.70.

- Save your model as ScanningMachines1VLProfit.s8
**Analysis Exercise**

**Exercise 1.** Add the cost and revenue information to the simulation model with the new scanning machines (*ScanningMachines2VL.s8*). Compare the two systems. Is one scanner going to give you better profits than the other? How do you show this statistically?

Save your model as *ScanningMachines1VLProfit.s8*

**Exercise 2.** Consider the Call Center model (*CallCenterVL.s8*). Account information personnel are paid $15 per hour. Technical support personnel are paid $17.50 per hour, while their managers are paid $25 per hour. Trunk lines cost $89 per week each. We do not know the revenues from the calls, but we do know that waiting on hold is annoying. Hold time costs are estimated to be $1.67 per minute for account information, $3.72 per minute for credit increase requests and $1.58 per minute for technical support calls.

Include the costs in the Call Center model and generate a 95% confidence interval on the total cost.

Save your model as *CallCenterVLProfit.s8*
4 Simulation Modeling with Arena

For the preceding four modules, we have used Simul8 as our modeling tool. Simul8 is a popular package that is easy to use. There are other packages used in manufacturing environments, such as AutoMod and ProModel. However, the package most commonly used in business settings is Arena.

Arena uses a flowchart representation of workflow through a system. In general, the ideas are similar to using Simul8, but the organization is a little different. This is because Arena is built on top of a long-lasting simulation language called SIMAN. In essence, Arena represents SIMAN programming graphically and puts together commonly used combinations of blocks and elements from SIMAN to make an easy to use drag-and-drop simulation package.

Let’s take a look.
4.1 Let's Start Arena

Start Arena. Go the Start button and find the Rockwell software program group. You should find Arena.

Start > Programs > Rockwell Software > Arena > Arena 5.0

It takes a little while to come up because it goes off and looks for a network license.
Standard Windows Menus

The Standard Windows Menus include the File, Edit, Windows and Help menus that are common to all Windows programs. The Tools, Arrange, Object and Run menus are specific to Arena.

- The Tools menu can be used to access other programs.
- The Arrange menu has options for aligning, grouping and rotating objects in the Model Workspace.
- The Object menu has options for connecting modules in a model.
- The Run menu contains run control and debugging commands.
Toolbars

The Toolbars include a Standard, a Draw and an Animation toolbar.

- The Standard toolbar contains the most commonly used commands from the menus, such as opening and closing models, cutting and pasting, connecting models and running, pausing and stopping model runs.
- The Draw toolbar contains tools for drawing backgrounds for the models you build.
- The Animation toolbar has tools for animating a clock, displaying a variable, plotting a graph and animating a simulation object, such as a queue and a resource.
Module Panels

The Module Panels contain modules for building basic models, advanced models and modeling advanced transfers, as well as the blocks and elements panels that contain low-level modules.

The Basic Process Panel contains flowchart modules that create, process and destroy entities in the model and spreadsheet modules that contain the parameters for entities, resources, queues and variables. The first model we build will use just modules from this panel. As we build more complex models, we will move on to the more advanced panels.
## 4.2 Arena’s Building Blocks

<table>
<thead>
<tr>
<th>Basic Process Panel</th>
<th>Type of Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create</td>
<td>Flowchart Modules: control the flow of entities through a model.</td>
</tr>
<tr>
<td>Dispose</td>
<td>Spreadsheet Modules: contain data about objects and variables in the model.</td>
</tr>
<tr>
<td>Process</td>
<td>We will use the Basic Process modules to build the scanning machines simulation in Arena.</td>
</tr>
<tr>
<td>Decide</td>
<td>We review the Arena modules in the same order as their equivalent Simul8 objects.</td>
</tr>
</tbody>
</table>
| Batch               | • **Create Module**  
| Separate            | • **Process Module**  
| Assign              | • **Dispose Module**  
| Record              | • **Resource Spreadsheet**  
| Entity              | • **Queue Spreadsheet**  
| Queue               | • **Work Entry Points**  
| Resource            | • **Work Centers**  
| Variable            | • **Work Exit Points**  
| Schedule            | • **Resources**  
| Set                 | • **Storage Bins**  

The simplest building blocks in Arena are contained in the Basic Process Panel. Arena has two types of modules:

- Flowchart Modules: control the flow of entities through a model.
- Spreadsheet Modules: contain data about objects and variables in the model.

Let’s go through the basic options for each object.
Create Modules

The applications will arrive in the model through a Create Module. Create Modules need to be told how often to create entities (applications) and how many.

- **Name** – the Create module.
- **Entity Type** – is more useful in Arena than in Simul8.
- **Time Between Arrivals** – are the inter-arrival times.
  - **Type** – specifies the probability distribution that represents the values the inter-arrival times can take and how likely each value is.
  - **Value** – appears when Exponential is chosen for the Distribution. Change the Distribution and the parameters to be filled in will also change.
  - **Units** – allows you to specify the units your parameters assume.
- **Entities per Arrival** – specifies the number of separate entities that arrive each time an arrival occurs. This can be random.
- **Max arrivals** – allows you to set a limit on the number of entities arriving through this module.
- **First Creation** – makes the first entity arrive at the beginning of the simulation run if checked. Otherwise a sample will be drawn from the Inter-arrival time Distribution.
We will model each scanning machine using a Process Module. It will delay the applications for the required amount of time and only take one at a time. Process Modules need to be told how long the work will take for each entity arriving to be worked on. Options are available for requiring resources for work to be performed.

- **Name** – the Process Module.
- **Delay** – specifies the time taken to perform the work on one entity.
  - **Delay Type** – specifies the probability distribution that represents the values the work times can take and how likely each value is.
  - **Units** – allows you to specify the time units.
  - **Allocation** – specifies the value of the processing activity. This is used to determine appropriate costs for the delay.
- **Report Statistics** – turns on statistics collection for this module if checked.
Simulation models need somewhere to dispose of finished work. The applications will finish up at the Dispose Module, allowing Arena to calculate statistics, such as the total time in the system.

- **Name** – the Dispose Module.
- **Report Statistics** – turns on statistics collection for this module if checked.
Resources

Resources work the same way in Arena as they do in Simul8. The Resources Spreadsheet needs to be told what the resource is called and how much of it there is. It can also be told how many of this type of resource is available, whether the availability of the resource change over time (according to a schedule) and whether it belongs to a set of pooled resources.

- **Name** – the resource so you can use an identifiable name to refer to it in the model.
- **Type** – specifies whether the resource has a fixed capacity or operates on a schedule (shift in Simul8). Schedules are specified in the Schedule module in the Basic Process Panel.
  - **Capacity** – specifies the number of entities that a resource is capable of handling at one time.
- **Busy/Hour, Idle/Hour, Per Use** – are costs.
- **Report Statistics** – turns on statistics collection for this resource if checked.
A queue is the place where the applications wait until the scanning machine is ready for the next application. Queue modules need to be told how the queue works. They can also be told how many items are stored at startup and what results should be collected.

- **Name** – the Queue.
- **Type** – is the operating policy for the queue. The choices are FIFO, LIFO, Lowest Attribute First and Highest Attribute First.
- **Shared** – specifies that multiple Process modules can share this queue.
- **Report Statistics** – turns on statistics collection for this resource if checked.
Let’s build the simplest model.

Step 1 – Drag and drop a Create Module onto the Model Window.

Step 2 – Drag and drop a Process Module onto the Model Window.

Step 3 – Drag and drop a Dispose Module onto the Model Window.

Unlike Simul8, we do not explicitly model the queue. Instead, when capacity of a Process Module is limited by requiring it to seize a resource, a queue is created and associated with the Process Module.

Notice that the objects are automatically connected with Route arrows to show the flow of entities. If the automatic arrows are wrong, click on the arrow and press the Delete key.

- Select a Route arrow and press the *Delete* key.
- Select the Connection Mode.
- Click on the small triangle to the right of the first object you want to connect.
- Click on the small block on the left of the second object you want to connect.
Specifying the Create Module

Double click on the Create Module.

- Name the Create Module *Arriving Applications*
- Type *Application* for the *Entity Type*
- For the *Time Between Arrivals*, choose *Random(Expo)* for the Type
- Change the *Value* to 6 and specify the *Units* as *Seconds*
- Click OK

When you type in a new Entity Type name, the type is created and can be edited in the Entities spreadsheet.

Are there any other options you think we should be using?
Specifying the Process Module

- Double click on the Process Module

- Name the Process Module Scanning Machine
- For the Action, choose Seize Delay Release
- Press Add and a resource dialog will appear
- Type Scanner for the Resource Name and press OK
- For the Delay Type, choose Triangular
- Change the Minimum to 20, the Most Likely to 25 and the Maximum to 30
- For the Units, choose Seconds
- Press OK

Notice that when you added the resource to be seized, a queue appeared over the Process Module.

Are there any other options you think we should be using?

---

Quantity specifies the number of units of a resource to seize.
Specifying the Dispose Module

- Double click on the Dispose Module

![Dispose Module dialog box]

- Name the Dispose Module Finished Applications

- Click OK
Take a Look at the Spreadsheets

It is always a good idea to see what has been automatically created in the spreadsheet modules.

Click on the Entity Spreadsheet.

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Initial Picture</th>
<th>Holding Cost / Hour</th>
<th>Initial VA Cost</th>
<th>Initial IWVA Cost</th>
<th>Initial Waiting Cost</th>
<th>Initial Toss Cost</th>
<th>Initial Other Cost</th>
<th>Report Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Picture Report</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>P</td>
</tr>
</tbody>
</table>

Double-click here to add a new row.

The Report picture looks like a document, so this works well. Do we have any cost information to enter? Let's hold off on this for now.

Click on the Queue Spreadsheet.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Shared</th>
<th>Report Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scanning Machine</td>
<td>First In First Out</td>
<td></td>
<td>P</td>
</tr>
</tbody>
</table>

Double-click here to add a new row.

This seems good.

Click on the Resource Spreadsheet.

<table>
<thead>
<tr>
<th>Name</th>
<th>Capacity</th>
<th>Type</th>
<th>Busy / Hour</th>
<th>Idle / Hour</th>
<th>Per Use</th>
<th>Status / Set Name</th>
<th>Failures</th>
<th>Report Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scanner</td>
<td>5</td>
<td>Fixed Capacity</td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0 now</td>
<td>P</td>
</tr>
</tbody>
</table>

Double-click here to add a new row.

How many Scanner resources are there? We can either say that the scanners have a capacity of five or that there are five scanning resources with different names. We will keep it simple and set the Capacity to 5. We could come back and use Failures later.
4.4 Running the Model

Your model should now look like this.

Press Run on the toolbar

The simulation run can be controlled from the toolbar. The buttons are made to imitate a tape or CD player.

You can slow the animation down by typing < (shift and “,”). You can speed it up using >.

You can run without animation by pressing instead of . The simulation seems to be running nicely. The queue grows appropriately.

Within the model, the default time unit is hours. Let’s change it to seconds.

Select Run > Set-up

Choose the Run Replications tab

Change Base Time-Units to Seconds

Now we can draw a graph with the same axes we used in Simul8.
Adding a Plot

Plots and other types of graphs are available on the Animate toolbar.

- Press Plot on the toolbar
- Press Add next to the Expressions list
- Right-click in the Expression box and choose Build Expression

The Expression Builder will appear.
Expression Builder

The Expression Builder dialog can be accessed in a similar manner from many dialogs and can be used to enter any formula from simple to complex. Through it you can access almost any quantity in the model.

- Expand the Basic Process Variables
- Expand Queue
- Choose Current Number in Queue

Under the Queue Name, you can specify which queue you mean. We only have one. Under the Current Expression, you can see the SIMAN code that represents this formula. Before version 5 of Arena, you had to know all of these SIMAN functions.

- Press OK

Let’s finish the plot.
Finishing the Plot

Arena is not very good at scaling plots, so we need to specify the range on the axis.

Type \(10\) for the Maximum

This refers to the y-axis on the plot.

Press OK

Type \(500\) for the Time Range\(^7\)

Check X-Labels

Press Run \(\rightarrow\)

What happens after a while?

The # History Points is the number of \((x, y)\) values that are saved in memory. Once the 100 points are full, the oldest are discarded.

Type \(10000\) for the # History Points

Don’t worry; we have plenty of memory. Re-run the model and take a look at the graph.

\(^7\) What are the time units when we can’t specify them? Select Run > Set-up and choose the Run Replications tab. The Time-Units default to Hours.
4.5 Analyzing the Results

To analyze the full model, we need to determine a warm-up time, a run length and a number of replications to perform.

- Select Run > Set-up
- Choose the Run Replications tab.

![Run Setup](image)

We can modify these values here. Right now there is no warm-up and we are performing one endless replication.
**Determining Warm-up**

We can use the plots to determine the appropriate Warm-Up Period. However, if you run the model multiple times, you get the same stream of random numbers and thus the same results. Not much use!!

In the Run Replications tab

- Type 10 for the Number of Replications
- Type 500 for the Replication Length and change the Time Units to Seconds

Go to the Run Control tab

- Check Pause Between Replications
- Press Run

Arena will run one 500-second simulation, then pause. You might want to speed up the simulation (>). You have to press Run again to run the next replication. The next replication will use a different stream of random numbers.

Again, 100 seconds looks like it will work.

- Type 100 for the Warm-up Period and change the Time Units to Seconds
Warm-up and Replication Length

We have specified 100 seconds for the warm-up. In Simul8, we collected statistics for 500 seconds. Arena works a little differently in this respect.

Warming Up

Arena specifies the Warm-Up Period and the Replication Length as overlapping times.

- Warm-Up Period
- Simulation Time
- Replication Length

Type 600 for the Replication Length and change the Time Units to Seconds
Analyzing the Results

We have already specified that we want 10 replications, so Arena will automatically give us confidence intervals. Running multiple replications is the same as a trial in Simul8.

- Un-check Pause Between Replications
- Press Fast-Forward
- At the end of the runs, choose Yes to see the results

The reports feature in Arena gives a summary of the replication results.

I have expanded some of the results categories. You can click on an item to see the corresponding page.

Are the results different obtained in Arena different from the results obtained in Simul8?
**Routes versus Connections**

In Simul8, when an entity moves between objects time is expended. In Arena, this is not the case. In Simul8, the arrows are routes. In Arena, we are currently using connections. Connections just show the flow of entities through the system and are cleverly animated to show movement along them. They do not, however, affect the statistics by expending simulation time. We can use routes in Arena, but this is a little more complicated and will be left until later.

 tasarım

Save your model as ScanningMachines1.doe
Modeling Exercise

Model the scanning process with the new scanners in Arena.

Save your model as `ScanningMachines2.doe`
**Costs in Arena**

In Simul8, revenues and costs are specified for each object, Work Entry Points, Storage Bins, Work Centers, Resources and Work Exit Points. When entities reach the object and certain activities are completed, cost and revenue totals are updated.

Arena works rather differently; there are no revenues and costs revolve around the entities and the resources. Resources can be assigned different costs per time unit values for busy periods and idle periods and they can be assigned per usage costs.

Entities can be assigned all sorts of different costs per unit time. Activities in the simulation, such as waiting in queues, being processed or being transferred from one place to another, can be defined as value added, non-value added, waiting or transfer activities. While entities are involved in different categories of activity, the corresponding cost category is accumulated for that entity. As resources are seized to process an entity, the resource usage costs are added to the costs for that entity too.

As entities leave the system, there various categories are added to the overall total for the various categories.
**Costs in Arena**

In our case, because we want to accumulate revenue for scanned applications, we will have to use an output that calculates the required expression. Here are the steps to add the cost information.

- Open the Resource Spreadsheet (Basic Process Panel)
- Change Busy / Hour to 0.1833 ($20 per day)
- Change Idle / Hour to 0.1833 ($20 per day)

The scanner costs the same whether it is busy or idle.

- Open the Statistics Spreadsheet (Advanced Process Panel)
- Add a new row called Revenue and set the Type to Output
- Build the following expression: \( \text{EntitiesOut(Application)} \times 0.01 - \text{ResIdleCost(Scanner)} - \text{ResBusyCost(Scanner)} \)

This takes the number of scanned applications multiplied by 1¢ per scanned application and subtracts the total resource idle cost and the total resource busy cost for the scanner. The statistic will show up in the Results Reports under User Specified.
Analysis Exercise

For both the model of the old scanning machines (ScanningMachines1.doe) and the model of the new scanning machines (ScanningMachines2.doe), add in the cost information. Run the two models for the total replication length of 600 seconds, with a 100 second warm-up period. Compare the two systems.

Save the models ScanningMachines1Costs.doe and ScanningMachines2Costs.doe
Now let’s try a more complicated model. We will model the Call Center.

We have a number of new modeling issues that need to be handled in this model:

- **Attributes and Assign Modules** — We need to be able to differentiate between the call types and product types, so the entities need to have attributes.
- **Sub-models** — This model is complex enough that it will be worth breaking the model up into pieces. The overview could be shown to managers and the detail kept for the analysts.
- **Scheduled Arrivals** — The time between arrivals does not stay constant over a day.
- **Decide Modules** — The entities will not follow simple paths through the model. We will have to control their destination when leaving each part of the model.
- **Scheduled Resources** — The availability of resources will vary over the day.
- **Sets of Resources** — The resources capable of handling the work depend on training. We will need groups of resources that can be called upon for each product type.
5.1 Attributes and Assign Modules

Attributes are a useful modeling tool in Arena. They are descriptions of the entities in a model that are passed along with the entities and can be accessed by the objects in your model. Their purpose is exactly the same as labels in Simul8:

- Controlling Work Flow
- Entity Specific Processing Times
- Segregating Results
- Prioritized Queues
- Entity Specific Shelf Life
Creating Attributes

To create a label:

- Add an Assign module to the model window.

Assign modules are used to make assignments to attributes.

- Double-click on the Assign module.

You can name the assignments and add and edit assignments to variables or entities’ attributes.

- Press Add.

- Change the Type to Attribute

- Type Call Type for the Attribute Name.

We now need to specify the distribution.
Specifying Distributions

The caller first hears a recording and is asked to press one for account information, two for credit increase requests and three for technical support. So we will set the Call Type to 1 for account information, 2 for credit increase requests and 3 for technical support. About 76% of customers request account information, 16% request a credit increase and 8% request technical support.

To create this distribution we actually use the Expression Builder again.

- Right-click in the New Value box and choose Build Expression.
- Select Random Distributions for the Expression Type.
- Select Discrete Probability.
- For the Cumulative Probabilities and Values type 0.76, 1, 0.92, 2, 1.0, 3.
- Click on a bar of the probability plot and drag it up and down.
- Press OK.

Why are the probabilities 0.76, 0.92 and 1.0.
5.2 Sub-Models

To break up a complex model, sub-models can be used to separate the pieces. For instance, we could have the following sub-models:

- Call Arrivals;
- Account Information;
- Credit Increase Requests;
- Technical Support.

To create a sub-window you have a couple of choices. If you have not created any modules to put in the sub-model yet:

1. Choose Object > Submodel > Add Submodel.

If you have created the modules that should be in the sub-model:

1. Drag a box around the set of modules you want in the sub-window.
2. Choose Object > Submodel > Aggregate.

To open a sub-model, double-click on the sub-model block. To name the window and create entry and exit point:

1. Right-click on the sub-model block and choose Properties.
2. Type the Submodel Name.
3. Edit the Number of Entry Points.
4. Edit the Number of Exit Points.
5. Press OK.

Notice that sub-model windows take up the whole Model Window when opened. It is not possible to watch the sub-model run while also watching the main model flow like it is in Simul8.
5.3 Scheduled Arrivals

Arrival rates that vary over time are called Scheduled in Arena. They are set up using the Schedule option in a Create module. Add a new Create Module to the Model Window.

- For the Time Between Arrivals, choose Schedule as the Type.

An option will appear asking for a Schedule Name. Leave the Create Module and go to the Schedule Spreadsheet in the Basic Process Panel. Schedule 1 will have been created.

- Change the Name to Call Arrival Rates.
- Keep the Type as Arrival.
- Press the button under Durations.

You will get the Schedule dialog.

- Press Options, change the Y-Axis Maximum to 100 and press OK.

Change the arrival rates to the appropriate levels by clicking in the slots; blue bars will appear and their value is shown in the bottom right as Arrival Rate = xx.

The specified rates will be the average rates. The time between arrivals will still be exponential.

Make sure that the Schedule Name in the Create Module is Call Arrival Rates.
5.4 Decide Modules

We want to send the different call types to different parts of our simulation model. To demonstrate this method, add three exit points to the Call Arrivals sub-model.

- Add a Decide module and connect the Assign module to it.
- Double-click on the Decide Module.
- Change the Name to Separate Calls.
- For the Type, choose N-way by Condition.
- Press Add under Conditions.
- For the If, choose Attribute.
- For the Named, choose Call Type.
- For the Is, choose ==.
- Leave the Value as 1 for the first Call Type.

- Press OK.
- Add another condition for Call Type == 2.
- Press OK.

You should see three exit points coming out of the Decide Module. If you read really small the first is called Call Type == 1, the second Call Type == 2 and the third Else.

- Connect the exit points from the Decide Module to the exit points from the sub-model.
5.5 Scheduled Resources

The tech support personnel work according to shifts. Thus the availability of resources will change over the day.

- Go to the Resource Spreadsheet in the Basic Process Panel.
- Create a new resource and name the resource EM 1.
- Change the Type to Based on Schedule.

Again we need to specify a Schedule Name. First we have to create the schedule.

- Go to the Schedule Spreadsheet. Add a new schedule.
- Change the Name to Early Morning.
- Keep the Type as Capacity.
- Press the button under Durations.

You will get the Schedule dialog.

- Press Options, change the Y-Axis Maximum to 1, change the Time-slot Duration to 30 minutes and press OK.

Change the arrival rates to the appropriate levels by clicking in the slots.

Notice that we can easily model the half-hour break in Arena.
5.6 Sets of Resources

Go ahead and create 11 resources. There are four on the early morning shift. I called them EM 1, EM 2, EM 3 and EM M. EM 1 stands for early morning person trained on product 1. EM M stands for the early morning manager\(^8\). Remember there is a manager on the early morning and the late morning shifts and they are trained for all three products.

The module for product-one tech support calls is going to need to seize a person trained for product one. If it cannot, then the customer will have to wait in the electronic queue and listen to annoying music. However, the work center can seize any of the product-one trained personnel who might be working at that time. It can choose from a set of people.

- Go to the Set Spreadsheet in the Basic Process Panel.
- Add a new set to the model and name the set Product 1 Personnel.
- Keep the Type as Resource.
- Press the button under Members.
- Add a new member and use the drop down box to choose the appropriate Resource Name.
- Repeat until all appropriate resources are in the set.

In the Process Module, when specifying the resource to be seized:

- For the Type, choose Set.
- For the Set Name, choose Product 1 Personnel.
- Keep the Selection Rule as Cyclical.

---

\(^8\) You can copy objects easily in Arena by right clicking on the object and choosing Duplicate. Drag the duplicate module to its place in the Model Window.
Modeling Exercise

Build the Call Center model in Arena.

Save your model as CallCenter.doe
Analysis Exercise

Add in the costs to the Arena Call Center model. Recall that account information personnel are paid $15 per hour. Technical support personnel are paid $17.50 per hour, while their managers are paid $25 per hour. Trunk lines cost $89 per week each. We do not know the revenues from the calls, but we do know that waiting on hold is annoying. Hold time costs are estimated to be $1.67 per minute for account information, $3.72 per minute for credit increase requests and $1.58 per minute for technical support calls.

Save the model as CallCenterCosts.doe
Until now we have built simulation models with assumed distributions for times between arrivals and processing or delay times. Where did they come from?

A stochastic simulation is a computer program that takes as its input probability distributions, runs this input through the logic that represents how the system works and generates output statistics. Thus far, we have been discussing the logic part of the simulation model. In this section, we will discuss the input probability distributions.
6.1 Input Analysis with Data

The preferred method for determining input probability distributions is through the statistical analysis of data. As an example, we could collect the times that applications arrive at the scanners or the times that calls arrive at the call center. From these times, we can calculate the times between arrivals and draw a histogram of their distribution.

The next step is to determine which family of probability distributions would most closely represent the data. Here is the process that most simulation input packages follows:

1. Fit as many different potential probability distributions as the package allows.
2. Calculate a goodness-of-fit measure for each candidate distribution.
3. Rank the candidate distributions by the measure.
4. Pick the most suitable distribution⁹.

⁹ This does not necessarily have to be the top-ranked distribution.
6.2 Stat::Fit for SIMUL8

Stat::Fit is a third-party distribution fitting software that is supported by Simul8. Let’s first take a look at how we go about fitting distributions in Stat::Fit.

- Go to Start > Programs > Simul8 > Stat-Fit
- Select File > Open
- Change the Files of Type to Data Files (*.txt)
- Select the file Input1.txt and press Open

The data table shows that the file contains 50 values.
Examining Histograms

Let’s take a look at the histogram.

Select Input > Input Graph

You will get the following window:

What does the histogram tell you?
Ranking Candidate Distributions

What do the goodness-of-fit measures tell us?

- Select Fit > Auto::Fit

- Select assigned bound and type 0 for the Lowest Value Allowed

- Press OK

A ranking of Stat::Fit’s list of continuous distributions is generated.

<table>
<thead>
<tr>
<th>distribution</th>
<th>rank</th>
<th>acceptance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta(0.5, 3.21, 1.94, 2.11)</td>
<td>9%</td>
<td>do not reject</td>
</tr>
<tr>
<td>Weibull(0.1, 1.69, 2.78)</td>
<td>87.2</td>
<td>do not reject</td>
</tr>
<tr>
<td>Gamma(0.2, 2.01, 1.9)</td>
<td>73.3</td>
<td>do not reject</td>
</tr>
<tr>
<td>Pearson 6(0.7, 5.69e+095, 2.1, 1.50e+095)</td>
<td>73.3</td>
<td>do not reject</td>
</tr>
<tr>
<td>Erlang(0.2, 1.9)</td>
<td>73.3</td>
<td>do not reject</td>
</tr>
<tr>
<td>Triangular(0.5, 0.1, 0.351)</td>
<td>61.8</td>
<td>do not reject</td>
</tr>
<tr>
<td>Lognormal(0.4, 0.494, 0.853)</td>
<td>26.5</td>
<td>do not reject</td>
</tr>
<tr>
<td>Rayleigh(0.178)</td>
<td>25.7</td>
<td>do not reject</td>
</tr>
<tr>
<td>Chi Squared(0, 2.50)</td>
<td>8.18</td>
<td>reject</td>
</tr>
<tr>
<td>Exponential(0.2, 1.44)</td>
<td>0.876</td>
<td>reject</td>
</tr>
<tr>
<td>Pearson 5(0, 1.05, 0.996)</td>
<td>0.146</td>
<td>reject</td>
</tr>
<tr>
<td>Power Function(0.45, 0.833)</td>
<td>0.114</td>
<td>reject</td>
</tr>
<tr>
<td>Uniform(0.5, 0.38)</td>
<td>4.35e-003</td>
<td>reject</td>
</tr>
</tbody>
</table>
Comparing the Distributions

Can we compare the distribution to the data graphically?

Click on Beta

The fit looks pretty good. What about the other candidate distributions?

Click on Weibull and then Gamma

They all look pretty good.
Other Graphical Comparisons

A Q-Q (or Quantile-Quantile) plot is a graph where one axis shows the quantiles of the data, while the other shows the quantiles of the fitted distribution. If the data fits the distribution well, then these points should plot along a straight line. Deviations from the straight line indicate poor fit.

Select Fit > Results Graphs > QQ Plot
You will see the Q-Q plot for the top-ranked distribution.

In the distribution selection box, select Weibull and Gamma
This gives a comparison of the three distributions.

It looks like the Weibull fits better in the tails of the distribution, whereas the Beta fits better for the middle values.
Other Graphical Comparisons

A box plot is a figure showing the first, second and third quartiles of the data or of the distribution (25th, 50th and 75th percentiles, respectively). The box runs from the first to the third quartiles; this is called the inter-quartile range. The line in the middle indicates the second quartile (also called the median). The next lines out are the octile (one-eighth) points and the outermost lines are the extremes of the data. If the data fits the distribution well, then all these points should fall in line.

Select Fit > Results Graphs > Box Plot
You will see the box plot for the top-ranked distribution.

In the distribution selection box, select Weibull and Gamma
This gives a comparison of the three distributions.

Each distribution fits the interquartile range and the median quite well. The problems seem to come in the extreme values.
Goodness-of-Fit Tests

Notice that some of the distributions are indicated as do not reject, while others say reject. What does this mean?

It is possible to form a hypothesis test for goodness-of-fit. The null hypothesis is that the data conforms to the tested distribution. The alternative hypothesis is that there is another distribution that would fit better. Various tests use different test statistics:

- **Chi-square**: takes the difference between the height of each bar in the histogram and the value of the overlaid probability density function at the middle of the bar, squares the differences and adds them up.
- **Kolmogorov-Smirnov**: takes the largest deviation from the straight line in the Q-Q Plot.
- **Andersen-Darling**: takes the deviation between the candidate distribution’s cumulative distribution function and the empirical distribution function of the data, squares it and integrates over all possible values. Weights are applied to stress the tails of the distribution.

A more complete discussion of each test can be found in the Help system under automatic fitting.

To see the settings for these tests, go to Fit > Setup and choose the Calculations tab.
Goodness-of-Fit Tests

In general, I have some advice for all these goodness-of-fit tests. The fit and the tests are totally insensitive for fewer than 10 data points [Stat::Fit won’t respond to less data], and don’t achieve much accuracy until 100 data points or so. The tests work best with on the order of 200 data points. For large data sets, greater than 4000 data points, the tests can become too sensitive, occasionally rejecting a proposed distribution when it is actually a useful fit.

In general it is usually appropriate to look at the ranking, examine the top group of candidate distributions and pick the simplest. Why the simplest? Well it is entirely possible that as part of your study you might be asked what-if questions like:

- What would happen if we had a 50% increase in call volume?
- How would the system perform if the scanners broke down 25% less?
- What would the effect be if the scanners could scan 40% faster?

To answer these questions we need to be able to play around with the distributions. This is easier to do with simple distributions.

A useful feature in Stat::Fit is the distribution choice guides in the Help system.

🔗 Select Help > Help Topics

🔗 Under Contents, select Guides > Guide to Distribution Choice

🔗 Answer the questions to receive guidance
6.3 Arena’s Input Analyzer

Input Analyzer was created to support input distribution fitting in Arena. Let’s first take a look at how we go about fitting distributions in Input Analyzer.

- Go to Start > Programs > Rockwell Software > Arena > Input Analyzer
- Select File > New
- Select File > Data File > Use Existing
- Change the Files of type to Text Files (*.txt)
- Select the file Input1.txt and press Open

The report shows that the file contains 50 values.
Examining Histograms

Input Analyzer automatically shows the histogram

Basic descriptive statistics are also shown automatically.
Fitting Distributions

What do the goodness-of-fit measures tell us?

Select Fit > Fit All

The top ranked distribution is shown overlaid on the histogram and the goodness-of-fit tests are shown above the descriptive statistics for this distribution.
**Ranking the Distributions**

Let’s take a look at the ranking.

iero Select Window > Fit All Summary

The ranking is similar to that obtained in Stat::Fit, but not identical. The ranking here is done by squared-error or Chi-square measure of fit.

You can take a look at the other candidate distributions, by choosing the Fit menu and picking the distribution.
Other Graphical Comparisons

Input Analyzer does not give other graphical comparisons.
6.4 Input Analysis without Data

In some projects, the powers that be may want you to build a simulation to show the usefulness of the model, a proof of concept. In some cases you will be lucky and they will already have the necessary data. In most cases, people do not collect data with simulation models in mind. In cases where the data is not available, you must turn to individuals, experts familiar with the system.

Here are some simple questions you can ask them and the corresponding distributions you can use.
**Highest, Lowest and Most Likely**

Most people with sufficient knowledge about a random numerical quantity could tell you the most likely or most common value, the lowest possible value and the highest possible value. If they can give you these three quantities, you can use a triangular distribution.

---

**Quiz**

The neighboring republics of Turkmenistan and Uzbekistan were both formerly part of the Soviet Union. In 1990 Turkmenistan had a population of 3.7 million. Estimate Uzbekistan's 1990 population.
Anchor and Adjust

Uzbekistan's 1990 population was 20.7 million.

Anchor and Adjust

Given numerical information, we tend to start from there and adjust. Watch out.

Kahneman and Taversky performed an experiment in which subjects were asked to estimate various quantities, such as the percentage of African nations who were members of the United Nations. Before the estimates were elicited, a wheel of fortune was spun to generate random numbers between 0 and 100 and the subjects were asked whether the required percentage was above or below the generated number. They were then asked to make an estimate by moving up or down from the random number. The random number had a significant effect on the estimates given.

So how do you avoid this? Ask for the maximum value first, then the minimum and then the most common.
Other Techniques

Stat::Fit’s Help system gives some further advice for determining input distributions with no data.

Is the variable bounded?

- Unbounded
  - Use Normal Distribution

- Bounded above a minimum
  - Time to a random event
    - Use Exponential Distribution
  - Time to a complex event
    - Use Gamma Distribution
  - Time to task completion
    - Use Gamma Distribution
  - No further information
    - Use Uniform Distribution

- Bounded between a minimum and a maximum
  - Zero at or near both bounds
    - Use Triangular or Beta Distribution
  - Non-zero at one or both bounds
    - Use Beta Distribution

You can go through the full list of questions in Stat::Fit using the Guide in the Help system.

- Select Help > Help Topics
- Under Contents, select Guides > Guide to No Data Representations
- Answer the questions to receive guidance
**Analysis Exercises**

You will have to do these exercises in Arena’s Input Analyzer, as the student version of Stat::Fit is limited to 50 data points.

**Exercise 1.** Let’s take a look at fitting distributions with different amounts of data.

- Start a new file in Input Analyzer
- Select File > Data File > Generate New
- Select a Gamma distribution
- Set Alpha to 3, Beta to 2 and the Offset to 0
- Generate 25 points
- Determine the best distribution using Fit All
- Test the fit for a gamma distribution

Repeat the exercise with the same distribution for 100, 200 and 1000 points. Do your findings support my general recommendations?

**Exercise 2.** Let’s take a look at fitting bounded distributions with different amounts of data.

- Start a new file in Input Analyzer
- Select File > Data File > Generate New
- Select a Beta distribution
- Set Alpha1 to 3, Alpha2 to 2, the Minimum to 0 and the Maximum to 1
- Generate 25 points
- Determine the best distribution using Fit All

Repeat the exercise with the same distribution for 100, 200 and 1000 points. Do your findings support my general recommendations?
Simulation modeling can be very interesting. It is more fun than other forms of more mathematical modeling; there are pretty pictures. It sometimes seems like you are building your own video game.

The modeling process is a learning experience. By the time you are finished modeling a system, you really do know it inside out. You get a different perspective and a deeper understanding, even if you have worked in a system for a long time. The ability to play with a system at no cost (other than computer time) is very useful.

However, when it comes down to the bottom line, the knowledge is useful, but you built this simulation to help make decisions about improving the system. There are things about the system you can control to improve it; we will call these decision variables or controls. There are measures of performance of the system; we will call these responses and our objective is to improve them. There are probably constraints on how much we can change things, such as limits on space or budget. This is the language of optimization. We need to make decisions about the system and attempt to optimize the outcome.
7.1 Picking the Best Alternative

In some cases, our decision consists of a few choices. A small set of possible alternatives. Considerable work has been performed on the statistical techniques that allow us to pick the best alternative from a small set. This work is based on techniques such as Analysis of Variance, which you are probably familiar with. However, our goal here is a little different than one’s goals when using standard statistical tools. We do not want to know whether there are any differences between the average responses of the alternatives, or even which ones are different. We want to put all the information we have into determining which is the best one!

Arena comes with a tool called Process Analyzer that allows us to do just this. The techniques used in Process Analyzer are based on very sound theoretical work performed by Barry Nelson, Julie Swann, David Goldsman and Wheyming Song.

Suppose the manager of the scanning process did not have a new scanner available. He is currently using five older scanning machines. Is this the best number to use? Can we improve the process’s revenues?

- Open the model `ScanningMachines1Costs.doe`
- Make sure the Replication Length is 600 seconds, with a Warm-up Period of 100 seconds
- Select Run > Check Model to re-create the .p file that Process Analyzer uses

---

Launching Process Analyzer

There are two ways to launch Process Analyzer.

Select Start > Programs > Rockwell Software > Arena > Program Analyzer

Or, in Arena:

Select Tools > Process Analyzer

Now that you have Process Analyzer open:

Select File > New

Follow the direction and Double-click here to add a new scenario

In Process Analyzer, the alternatives are called scenarios. The Scenario Properties dialog is used to specify the Arena model file (*.p) that will be used in the scenario.

Press Browse

Find your file ScanningMachines1Costs.p

Press OK

Recall that the simulation model in this file runs for a total of 600 seconds, with the first 100 seconds as warm-up.

We need to specify our controls, responses and scenarios.
**Taking Control**

We will make the number of scanning machines a control and Select Insert > Control

- Expand the Resource list of Controls
- Select Scanner
- Press OK

The control is added and set to the default; its level in the original simulation model file.

A useful control to add every time is the number of replications. We may want to perform more replications than are specified in the Replication Parameters of the model.

- Expand the System list of Controls
- Select Num Reps
- Press OK

The default is again the number of replications specified in the simulation model file.
Watching the Response

The response will be our revenue statistic.

- Select Insert > Response
- Expand the User Specified list of Responses
- Select Revenue
- Press OK

The response Revenue is blank because no replications have been performed.
Creating the Scenarios

The last set-up step is to create the alternatives we want to run.

- Click on the 1 to the left of Scenario 1
- Right-click and select Duplicate Scenario
- Repeat until you have 10 scenarios
- Rename them Scenario 1 through Scenario 10
- Change the levels of the Control called Scanner to run from 3 to 12

We have our 10 scenarios with various numbers of scanning machines.

- Change Num Reps to 5 for each scenario

This will speed up the process.
Graphing the Results

To run the scenarios, select the rows of the scenarios you want to run and press run.

- Click on the 1 to the left of Scenario 1
- Hold down the Shift key and click on the 1 to the left of Scenario 10
- Select Run > Go
- Press OK in the dialog box that appears

This will take a while. You can see the count of the number of replications performed for each scenario.

Is there a definitive result here? More scanners cost more per day. However, more scanners can also scan more applications. There is a trade-off.
Graphing the Results

Can we say that the highest average response value indicates the best number of scanners to use?

- Select the Response column called Revenue
- Select Insert > Chart
- Select a Box and Whisker chart
- Press Next three times
- Check the box Identify Best Scenario
- For revenue, select Bigger is Better and press Finish

Here is the resultant chart.

![Chart Image]

We are 95% confident that the best scenario is contained in the set of all red scenarios! So the best choice is either 4 or 5 of the old scanners.

How can we get a more definite answer? How do you decrease the width of a confidence interval?
More Replications

We will increase the number of replications to 10. This will take a while, so you may want to follow along. Here are the steps:

- Change the levels of the Control called Num Reps to 10

You will be asked if you want to continue, as this will lose the data from the runs you have performed so far. It would be nice to only have to do 5 more replications, but oh well.

- Select Scenario 1 through Scenario 10 and run them
- Create the box and whisker chart again showing the best scenario

Here are the new results:

Scenario 2 with 4 scanning machines is the best alternative. Wow, the number we started with!!!
Analysis Exercise

Determine the best choice for the number of new scanning machines (ScanningMachines2Costs.doe).
7.2 Optimizing Your System

In many cases, you will not be able to keep your set of alternatives down to a small number. In this situation, you have controls that can vary within reasonable bounds and possibly constraints, such as the budget, space or resource constraints.

In finding the best number of scanning machines, I arbitrarily used alternatives with between three and ten scanners. How did I know the best value would be in this range? Answer: I already knew the best value!

I ended up running nine alternatives for 100 replications. Actually, I had already run all ten alternatives for 10 replications. This is a lot of computer time, even for a small simulation run for only 600 simulated seconds.

Is there a better way to do this?
Deterministic Optimization

Researchers and practitioners have accumulated a vast field of knowledge in mathematical programming or optimizing a known function with known constraints.

Suppose we have two controls and a single response. We want to vary the controls within reasonable ranges and find the highest possible value of the response. We could take the brute force approach and calculate a grid of values in the range allowed and calculate the response at each grid point.

The highest value can be obtained somewhere in the middle of the range. This involves lots of calculations of the value of the response.

We could do this faster by imagining a mountain climber. The mountain climber can start somewhere at the bottom of the mountain and head straight up the steepest slope. It can be shown that if the response function obeys certain mathematical properties, the mountain climber will eventually reach the highest point. This is called the method of steepest ascent. There are other methods that may reach the summit with fewer calculations.
Optimization with Random Responses

Suppose now that there is some underlying response function similar to the deterministic function, but now there is randomness in the response. Each time we calculate the response for given control values, we get a different response value. They still average out to the same value, but there is this annoying noise.

If there is a low level of noise, we might be OK.

Our mountain climber might be able to find the summit. However, remember that every time he turns around the terrain changes!!

In this case, we do not know the mathematical properties of the underlying average response function and we cannot guarantee that we will find the optimal value.
Heuristics to the Rescue

Heuristics are techniques for improving solutions that are approximate and self-educating, but they are not mathematically proven to find the optimal solution. They will usually find you a good solution though.

**Scatter search** is an information-driven approach, exploiting knowledge derived from the search space, specifically high-quality solutions found within the space and trajectories through the space during the search. Scatter search operates on a set of points, called reference points, which result in good solutions. The approach systematically generates linear combinations of the reference points to create new points.

**Tabu search** is then superimposed to control the composition of reference points at each stage. Tabu Search has its roots in the field of Artificial Intelligence. Memory is a fundamental concept in Tabu Search, which uses search history to guide the process. In its simplest form, memory prohibits the search from re-investigating solutions that have already been evaluated.
OptQuest

OptQuest is a generic program that implements scatter search and can be hooked in to many model programs. It is available in versions that hook in to Arena and Simul8.

The combination of scatter search and tabu search algorithms creates a highly effective solution process. The incorporation of such designs gives OptQuest the ability to solve complex simulation-based problems with unprecedented efficiency. However, the use of memory in OptQuest is much more complex than the basic tabu approach, using memory functions to encourage search diversification and intensification. These memory components let the search escape from the smaller peaks in the mountain range to find the summit of the highest mountain.
7.3 OptQuest for Simul8

OptQuest uses the same language as Process Analyzer, but it looks a little different. In this section, we will take the optimization approach to finding the best number of scanners in Simul8. Why would we do this? Well, remember that in the section on Simul8 Profit we set the average daily cost to $20 for the old scanners and ran a full day (86400 seconds) of simulation. If we were to run 10 scenarios for 100 replications of this length, we would be here all day. The optimization approach aims to minimize the total number of response function calculations. In simulation optimization that is the total number of replications.

First, we must modify the model.
Resources as Controls

In Process Analyzer, the software takes a look at the Arena model and gives you a list of all the modifiable parameters of the model for you to choose your controls or decision variables. OptQuest does the same thing. We want to set the number of available scanning machines as a decision variable. OptQuest does not make the number of replicates of a Work Center available as a decision variable. To overcome this problem, we will create fifty replicates of the Work Center, but require a scanner resource to be available for each Work Center to accept an application entity. In this manner, we can vary the number of available scanners from 1 to 50.

- Open the ScanningMachines1VLProfit.s8
- Change the number of replicates of the Scanning Machines Work Center to 50
- Add a new resource called Scanners
- Change the Number of this type of resource available to 5
- Right-click on the Number of this type of resource available to add it to the Results Summary

With the number available on the Results Summary, OptQuest will make this quantity available as a decision variable. Make sure that this quantity and the Profit from the Income Statement are on the Results Summary.

- Add the Scanners resource to the list of Resources Required in the Scanning Machines Work Center

We also need to modify the financial settings, so that each Resource costs $20 per day, not the Work Center.

- Set the Scanning Machines Work Center’s Capital Cost to 0
- Set the Scanners Resource’s Capital Cost / Resource Unit to 20
- Set the Scanned Application Work Exit Point’s Revenue (per Unit) to 0.01
Specifying the Decision Variables

Now we can start OptQuest.

Select Trials > OptQuest for Simul8

When OptQuest starts up, it accesses the model you have open in Simul8.

Select File > New

Select Tools > Wizard

The OptQuest Wizard will take you through the steps to set up an optimization.

Deselect all variables but Scanners.Number Available

Set the Lower Bound to 1

Set the Suggested Value to 10

Set the Upper Bound to 50

Press OK
Specifying Constraints

Constraints are limits on the decision variables. OptQuest requires simple upper and lower bound constraints on each decision variable. However, there may be other constraints. As an example, suppose we have three types of resources, the decision variables are the number of each type available and we can only use a total of fifty resources of all types.

In this case, we could specify that the sum of numbers available for each type should be less than or equal to 50. Using the list of decision variables on the right-hand side of the Constraints window, we could type the following constraint:

\[ \text{Scanners1.Number Available + Scanners2.Number Available + Scanners3.Number Available} \leq 50 \]

OptQuest would then only consider solutions that meet this constraint.

In our simple optimization, there are no constraints.

Press OK
Specifying the Objective

In an optimization, our goal is to maximize or minimize a quantity that is affected by changing the decision variables, while only considering values for the decision variables that stay within the constraints. OptQuest offers a list of possible responses in the model that could be maximized or minimized.

- Scroll down the list to Simulation Total.Total Profit on Income Statement
- Click on the drop down arrow in the Select column

You have three choices for each possible response.

| Minimize Objective | Maximize Objective | No |

Clearly, we want to maximize our profits.

- Select Maximize Objective
- Press OK
Specifying the Settings

The settings allow you to fine tune your optimization. The first set of options specify the time you want OptQuest to keep looking for the best solution. We can leave this at 10 minutes.

The second set of options specifies the number of trials per simulation and the other output options.

Press OK

You will be asked if you want to run the optimization now.

Press Yes
Running the Optimization

At the end of the optimization procedure, your results might look something like this:

The progression of the best solution thus far is shown in the graph. 50 simulations were performed at 50 different values of the decision variable. The table shows the simulations at which the best solution changed and the simulation with the overall best solution. Next to each simulation shown in the table are the value of the objective obtained and the value of the decision variable used in that simulation.

Notice that I gave OptQuest enough time to run 50 simulations. It found the best solution after 47.
**Simulation Accuracy**

OptQuest is still attempting to climb our mountain with ragged and ever changing terrain. The combination of tabu and scatter searches helps because they have a memory of what the terrain looked like the first time the mountain climber saw it. We can give it some help, however. We can smooth out the terrain.

OptQuest is using the average of 3 runs (replications) per simulation (trial). We can modify this in the options.

- Select Tools > Options
- Change the Number of Trials per Simulation to 5
- Re-run the optimization

The average of 5 values is less noisy than the average of 3 values. Here is what I got:

<table>
<thead>
<tr>
<th>Simulation</th>
<th>Simulation Total Profit (Income Statement)</th>
<th>Scanners (Number Available)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>56,9200</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>14,6400</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>43,0100</td>
<td>5</td>
</tr>
<tr>
<td>Best</td>
<td>58,2000</td>
<td>4</td>
</tr>
</tbody>
</table>

The best solution was found after 11 simulations; we really didn’t need the rest.
Analysis Exercise

Determine the best choice for the number of new scanning machines (ScanningMachines2VLP\text{\texttt{rof}}.s8).
7.4 OptQuest for Arena

OptQuest for Arena is virtually identical to the Simul8 version. The main changes are in the names Arena gives in the list of Controls (Decision Variables in OptQuest for Simul8) and Responses (Objectives in OptQuest for Simul8).

- Launch Arena and open the file ScanningMachines1Costs.doe
- Set the Replication Length to 600 seconds and the Warm-up Period to 100 seconds
- Select Tools > OptQuest for Arena
- Select File > New
- Select Tools > Wizard
- Select Scanner as a Control
- Set the Lower Bound to 1, the Suggested Value to 10 and the Upper Bound to 50
- Press OK
- Leave the Constraints blank and press OK
- Scroll down the list of Responses to find Revenue
- Select Maximize Objective for Revenue and press OK
- Press Yes to Run Optimization Now
Optimization Results

The results look very similar. Obviously the revenues are smaller because we are only running 500 seconds of simulation rather than a full day. This also makes the time to find the solution much faster.

Notice, however, that the optimal choice is still 4 scanners, whether we run 500 seconds in Arena or 24 hours (86,400 seconds) in Simul8.
Analysis Exercise

Determine the best choice for the number of new scanning machines (ScanningMachines2Costs.doe).
Simul8 Corporation has supplied the following information.
8.1 SIMUL8 Professional Services

After completing the VCU course in simulation Capital One analysts will be able to use SIMUL8 to construct a basic simulation and output associated measures including cost & revenue measures. It will also pass an understanding about how to perform optimization experiments. All of these skills contribute to the successful completion of a simulation project. It should also give the analyst the skills to identify projects suitable for simulation analysis. Never the less the analyst is likely to require some handholding while conducting the simulation analysis. In particular the initial simulation model design, a key stage in the simulation, can be difficult for an analyst new to simulation.

SIMUL8 provide a range of simulation services which include the following:

- Training
- Consulting
- Mentoring
- Central Service Support
Mentoring

The key service that will aid analysts in completing their simulation analysis is mentoring. This is a service that is designed to help people learn how to apply SIMUL8 to real projects. Our mentors can guide you through the entire project from start to finish or jump-start the project by building the first simulation. In mentoring analysts learn first-hand how to manage a simulation solution through the analysis, structuring, development, and testing phases. As the project proceeds the skill-transfer naturally occurs and the analyst completes more and more of the work un-aided. At the end of the mentoring program the analyst has a completed project and has added to their skill set knowledge for use on future projects.

Mentoring is very flexible and will be tailored to match you. However for the program to be successful there must be regular contact between you and the mentor. We recommend daily contact if possible.

It can be delivered onsite or remotely using email, telephone and web conferencing. Most customers opt for a combination of both methods. The cost depends on how much contact you wish with the mentor and how much of the program is delivered onsite. Fixed prices are agreed before the project begins.
Contact Details

SIMUL8 Corp
2214 Rock Hill Road, Suite 501
Herndon
VA 20170
Tel: 800 547 6024
Fax: 800 547 6389

Simon Johnson
Simulation Consultant
simon.j@simul8.com
Direct: 703 579 1096
8.2 Example Models in Simul8

Simul8 has provided some examples of models that might be pertinent to Capital One’s operations.
Staff Requirements in a Service Center

Custom results showing detail of customer wait times

Standard Results screen showing confidence of results after a trial.
Detail of a Teller Process

Dashboard:
- Timeliness: 32.2%
- Accuracy: 96.9%
- Sigma Score: 1.02

Microsoft Excel - banking_hall_data.xls:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td></td>
<td>Customer</td>
<td>Teller</td>
<td>Time</td>
<td>Accurate</td>
<td>Sigma</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>08:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>09:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>10:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>11:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td>12:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td>13:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>14:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td>15:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>16:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td>17:00</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Microsoft Excel - banking_hall_data.xls:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical To Quality Issues</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timeliness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service within 5 minutes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cash Counted Correctly</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low Stress</td>
<td>0.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>High Stress</td>
<td>10.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 18 June, 2004 by Jason R. W. Merrick, All Rights Reserved.
Detailed Back Office Processing