Simulation Modeling & Analysis with Arena ®

A Business Student's Primer

FIRST EDITION: JUNE 2005

SELECTED CONTENT FOR BU275

LAURIER
BUSINESS &
ECONOMICS

School of Business & Economics,
Wilfrid Laurier University

Justin Clarke

Simulation Modeling & Analysis with Arena ®:
A Business Student's Primer

Justin Clarke

Copyright © 2005

School of Business & Economics,
Wilfrid Laurier University
75 University Avenue West
Waterloo, ON, N2L 3C5

Warning and Disclaimer

This book consists of various basic and intermediate level simulation modeling examples in Arena® 7.01. All examples have been devised using the Basic Edition of Arena® 7.01 and are intended for academic purposes only.

This book and its components are provided to enhance knowledge and encourage progress in the academic community and are to be used only for research and educational purposes. Any reproduction or use for commercial purpose is prohibited without the prior express written permission of the primary contributors.

The author and the contributors make no representation about the suitability or accuracy of the examples provided herewith for any specific purposes and also make no warranties, either express or implied, or that the use of this book will not infringe any third party patents, copyrights, trademarks, or other rights. The examples and constituent data are provided "as is". The author, contributors, and School of Business shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the examples and models that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of the School of Business, Wilfrid Laurier University.

Feedback Information

The author and the reviewers of this book have created and tested the examples and models in this book with care and precision. It is our goal to create a primer of the highest quality and value, and your feedback as a reader will be a natural continuation of this process. If you have any comments on how we can improve the quality of this book, or otherwise alter it to better suit your needs, you can contact the author and/or the editor.

We greatly appreciate your assistance.

Author -- Justin Clarke: justin@webartpromotions.com

Editor -- Umar Ruhi: uruhi@wlu.ca

Trademark Acknowledgements

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. The author and reviewers cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

- Microsoft®, Microsoft Excel®, and Microsoft Word® are registered trademarks of Microsoft Corporation.
- Arena® is a registered trademark of Rockwell Software Inc.
- Mr. Lube® is a registered trademark of Mr. Lube

About the Author

Justin Clarke is currently a fourth year business student at Wilfrid Laurier University in the Honours Bachelor of Business Administration (BBA) program. He is specializing in Marketing and is pursuing a minor in Computer Science. Justin has served as President of PRISM, an information technology business that operates within Wilfrid Laurier University providing IT solutions to over 2500 members. He also runs a consulting business, WebArt Promotional Services, providing IT consulting to small businesses specializing in e-business solutions.

About the Reviewers

Alan Marshall is a Lecturer in Finance in the School of Business and Economics, where he has also frequently taught Business Decision Models and Operations Management. He has also taught at Atkinson College, where he was for 16 years, and the Schulich School of Business, both at York University. Mr. Marshall's professional experience has included three years at General Motors and running his own consulting firm, which has engaged with institutes, multi-national firms and a foreign government. He has also been a Provincial Examiner for the Society of Management Accountants and an instructor for both the Institute of Canadian Bankers and the Canadian Securities Institute.

Umar Ruhi is a member of the Operations and Decision Sciences Area and he lectures in Management Information Systems, Operations Management and e-Business at the School of Business & Economics. A software engineer by profession, Umar holds a B.Sc. degree in Computer Science and an M.B.A. in e-Business & Knowledge Management. With over six years of experience in e-Business consulting, Umar has managed operational technology projects through the design to the deployment stages, and consulted for multinational organizations such as General Electric, and Cisco Systems. His professional qualifications include various industry endorsed certifications including those from IBM, Cisco, Novell and Sun Microsystems. Additionally, Umar has lectured in various information systems and e-Business courses at the college and university level around Canada, and he has also been a visiting research scholar at Johns Hopkins University in the U.S. Umar's research interests include Management Information Systems, Knowledge Management, Mobile Technology Applications, Community Informatics, and Supply Chain Management Information Systems.

Introduction to Simulation:

Understanding Processes, Distributions & Decisions

Table of Contents

1.0   Warnings

Do NOT save your Arena models on floppy disks. Floppy disks are a very fragile media and are bound to become damaged no matter how carefully you take care of them. Save your work frequently onto a more stable source such as a hard drive, network drive, or a USB key. It should be noted that USB keys aren't without their own problems – but they are much more reliable than floppy disks.

Do NOT save documents directly on the hard drives of public computers as those files are often deleted every time you reboot the computer. If the computer crashes and reboots, you will lose any files on the hard drive.

Do NOT save Arena models deep within the folder structure of your computer. There is a limitation in Crystal Reports (the reporting mechanism of Arena) that cannot handle exceptionally long file paths. In newer operating systems (Windows 2000 or Windows XP) if you save your Arena models under My Documents or the Desktop you may have problems. This is because the actual path to these folders is already quite long. For example, the path to the My Documents folder in Windows XP is "C:\documents and settings\user name\My Documents". If you were to save your model within a few folders under My Documents, it is likely you would run into this problem. A safe location to save these models is to create a folder called "Arena" directly on the C:\ drive and then create a folder for each model inside this folder.

Arena will not display reports if you run your simulation from a non-writeable media (such as a CD or a write-protected floppy/USB key), as it cannot write the reports. Also, Arena will not display reports if you run the simulation from inside a compressed folder (zip) for the same reason. Save your model to the hard drive or a writeable media before running the simulation.

2.0   Prerequisites

The "Simulating Distributions" section assumes you have some functional skills using Microsoft Excel. The sections that cover Arena do not require you to have any previous experience with Arena; however, reviewing some of the resources identified in the "Additional Resources" section would be helpful. The Arena sections do assume you have some theoretical knowledge of queuing systems.

3.0   The Simulation Process

Many students feel that they need to open Arena right away after being given a problem. However, there are many steps that need to be completed before you begin simulation in Arena. The following is a series of steps that should be followed when performing a simulation:

1. Understand problem/system
2. Be clear about project goals
3. Formulate model representation; identify assumptions; set level of detail; identify inputs & outputs
4. Collect data for inputs; fit probability distributions
5. Build model (Excel, Arena, Crystal Ball, …)
6. Verify computer model represents conceptual model (walk thru)
7. Validate computer model
8. Compare inputs/outputs from model with reality (test data)
9. Design experiment(s)
10. Run experiment(s)
11. Analyze results; draw conclusions; insights? Implications?
12. Document what you've done

You can see that Arena doesn't come into play until the 5th step. It is important to recognize that a fair amount of time needs to be spent understanding the system, choosing how detailed the simulation will be, validating your model, etc.

This workbook will largely cover steps 4-7 (collect data, build model, verify/validate model) as well as steps 10-11 (run experiment, analyze results), however, some time will be spent describing the thought processes that need to happen while **designing** a model to simulate in Arena. Modeling in Arena is much easier if you spend a significant amount of time understanding the situation before you ever open the program.

4.0   Simulating Distributions

This section will introduce you to simulating arrivals (i.e. people arriving at a business) using various distributions in Microsoft Excel and Arena. In the following examples it is assumed that someone has already collected and analyzed the data to determine which distribution applies and what the relevant variables are.

You'll find much more information on how Arena works when we visit the actual examples later in this module (Section 6.0: Example One). We won't define entities, processes, resources, etc. yet, as that will be covered in detail later. We will simply set up a "Create" module (to create entities) and analyze the reports to see how closely Arena simulated the distributions.

It should be noted that some distributions, such as Poisson, are used more often to simulate inter-arrival times, while other distributions, such as exponential, are used more often to simulate service times.

4.1   Introduction to RAND()

We will make extensive use of RAND() in the Microsoft Excel examples in this section. RAND() is a function in Excel that will randomly create a number between 0-1. This number is recalculated any time a change is made in Excel (or if you force an update using F9 on the keyboard).

Why is RAND() used?
Probability values are always between 0.0 and 1.0 (0% to 100%), therefore, RAND() is very helpful, as it creates a random number exactly within this range. We can treat this number as a probability and therefore use it as input into various distributions. Of course, you could always come up with the numbers yourself (as long as they are between 0.0 and 1.0) but if you want to model several hundred (or even thousand) random samples, using RAND() is a much quicker way of coming up with these numbers.

To practice:
   1. Open Microsoft Excel
   2. Click cell A1
   3. Type the following in the formula bar:
      =RAND()
   4. Note that a number appears in that cell that is between 0-1
   5. Hitting F9 will cause Excel to re-calculate the random number
   6. Copy/paste this formula (or use the drag bar) down to cell A5
   7. Note how the numbers are different, and change when you hit F9 – note also that they numbers continuously stay within the range 0-1

Your spreadsheet should look something like this (but the values will differ):

	A	E
1	0.834717	
2	0.64704	
3	0.346727	
4	0.008708	
5	0.861824	
6		
7		

Note: Excel will never actually calculate the number 1.0 when evaluating RAND(), however it will come very close (i.e. 0.99999999) which is, for most purposes, more than close enough.

4.2 "Freezing" RAND() values

Since RAND() is re-calculated any time a formula/cell/chart/etc. is updated in Excel, it can be difficult to come to any "permanent" conclusions when your data is constantly changing its values. If you ever want to "freeze" the values of RAND() so that they stay the same, follow these steps:

1. Select the data you want to "freeze"
2. Click Edit
3. Click Copy
4. Click Edit
5. Click Paste Special
6. Click Values
7. Click OK

These steps force Excel to copy/paste the data on top of itself, but instead of pasting the formula =RAND() again, it pastes the actual **values** that were calculated. Now your data will stay the same, no matter what changes you make. If you ever want the data to be recalculated again, simply replace the values with the =RAND() formula again.

4.3 Continuous Uniform Distributions

The formula to determine the value x of a continuous uniform distribution:
$$x = a + \left[p \times (b - a) \right]$$

Variables:
a is the minimum value of the distribution
b is the maximum value of the distribution
p is the probability (a value from 0-1)

A continuous uniform distribution is one where the range of values between a and b is uncountable. For instance, time units can be broken up further than just seconds; therefore the range of values is uncountable.

Example:
p=0.20
a=4
b=8
$$x = 4 + \left[0.20 \times (8 - 4) \right]$$

$x = 4.8$

Simulating in Excel:

Following the above example, let's assume that customers arrive at Joe's Pizza Shack every 4-8 minutes in a uniform distribution. We want to simulate the time that will elapse between each arrival. **Note:** a=4, b=8, p will be calculated.

1. Create an Excel spreadsheet that has the heading "Input" in cell A1 and the heading "Time Between Arrivals (minutes)" in cell A2.

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2					
3					
4					
5					

2. Under the "Input" column, type =RAND() into cell A2
3. Copy/paste/drag this formula down 9 more rows so that you have 10 rows of random input data (in the range 0-1)

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.2194				
3	0.914381				
4	0.25328				
5	0.742612				
6	0.177612				
7	0.301011				
8	0.634975				
9	0.854629				
10	0.583948				
11	0.078362				

4. Next, type the uniform distribution formula (shown earlier) into cell B2, substituting the a with 4, b with 8, and the p value with a reference to cell A2 (as it will contain the calculated probability for that row).

	B2		fx	=4+(A2*(8-4))	
	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.5879	6.351598			
3	0.228301				
4	0.275226				

5. Drag this formula down to the other 9 rows

1	Input	Time Between Arrivals (minutes)
2	0.2231	4.8924
3	0.746663	6.98665
4	0.784749	7.138998
5	0.597793	6.391171
6	0.376891	5.507563
7	0.480778	5.92311
8	0.89476	7.579038
9	0.894309	7.577236
10	0.255457	5.021829
11	0.436354	5.745418

You now have good estimates as to how many minutes will elapse between each of the next 10 arrivals. These are known as **inter-arrival times** and are the first step in simulating systems using Microsoft Excel. Of course, this example isn't limited to only ten rows – you could create hundreds of rows of data simply by dragging these formulas down further.
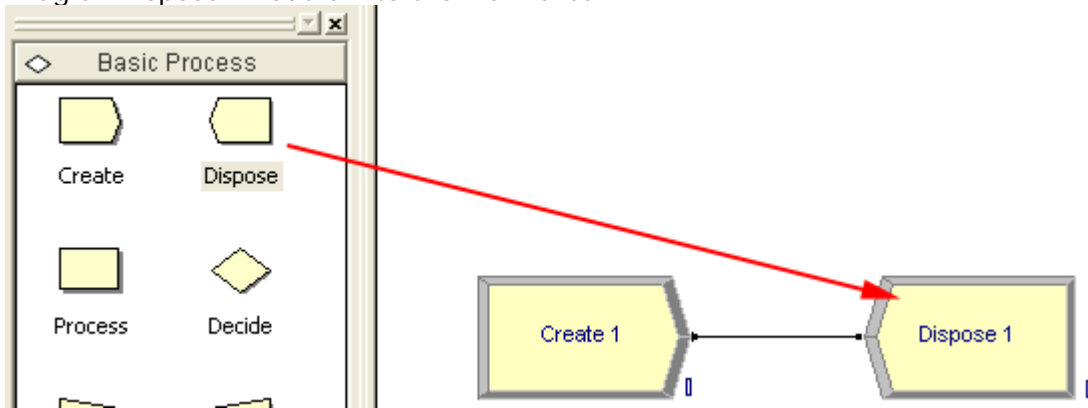
Simulating in Arena:

1. Open Arena
2. Save the model (preferably in its own folder)
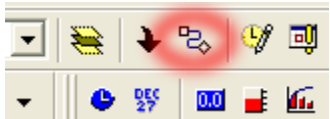3. Drag a "Create" module into the work area
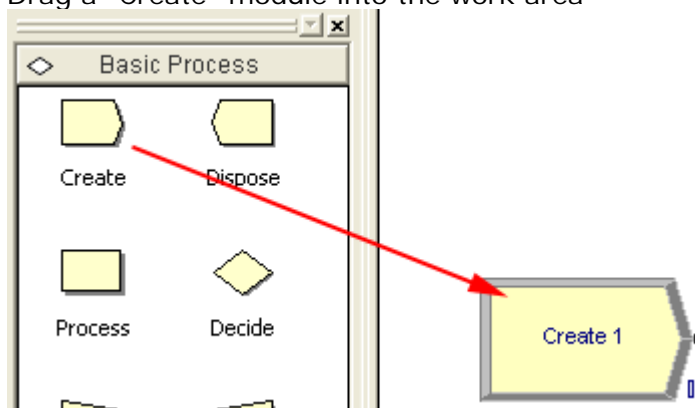
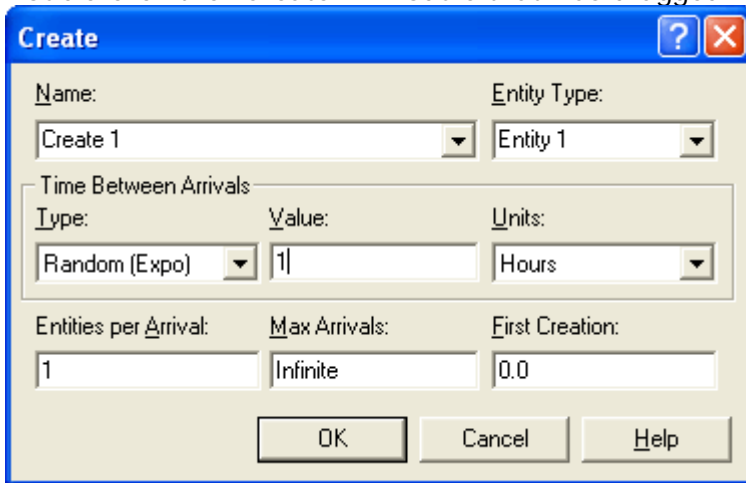4. Click the "Create" module (once) to make sure it is selected

5. Drag a "Dispose" module into the work area

Note: If your modules did not automatically connect, you will need to connect it manually. To do so, click the connect button (shown below) and click the two black dots on each module to connect them.

6. Double-click the "Create 1" module that was dragged in

7. We can leave "Name" and "Entity Type" as their defaults, as they are only there as labels.
8. Change "Units" to minutes, as our distribution (and variables) are defined in minutes
9. Entities per arrival = 1 (the number of entities that are created)
10. Max arrivals = Infinite (the population limit)
11. First creation = 0.0 (amount of time to elapse before creating the first unit)

12. Under "Type" choose "Expression"
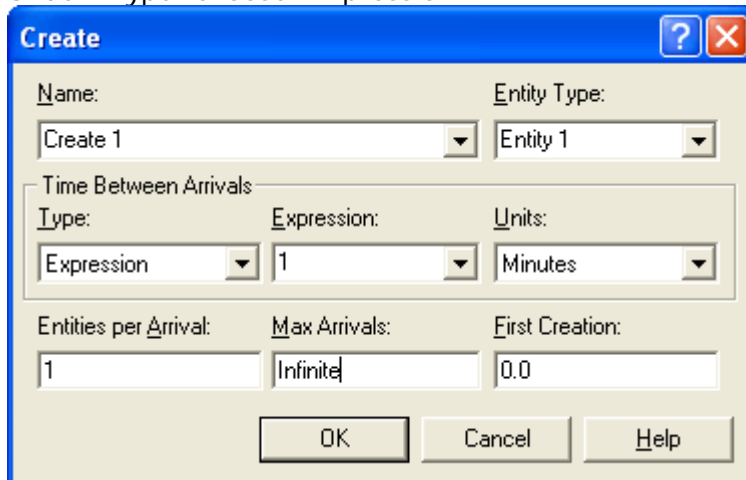
> **Note:** "Type" is the most important section of this module. This defines what distribution (and key variables) the "Create" module will use when creating entity arrivals.

13. Right-click anywhere in the "Expression" dropdown menu
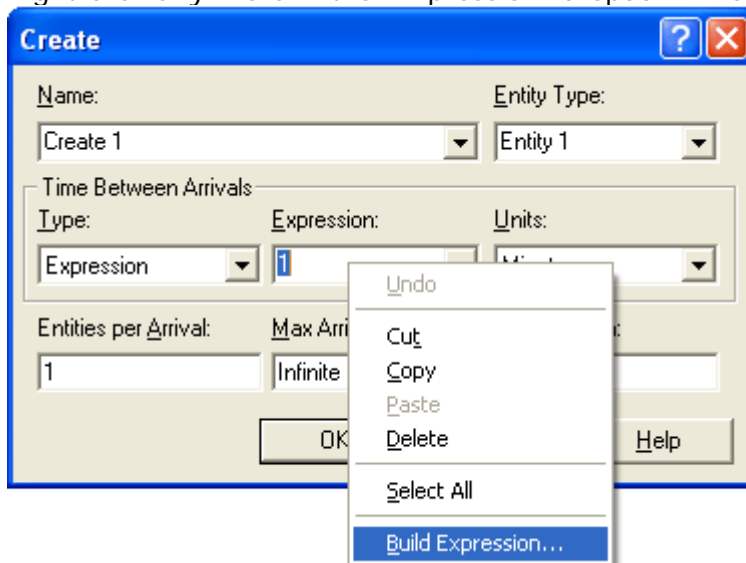
14. Click "Build Expression"

15. Click the + beside "Random Distributions"
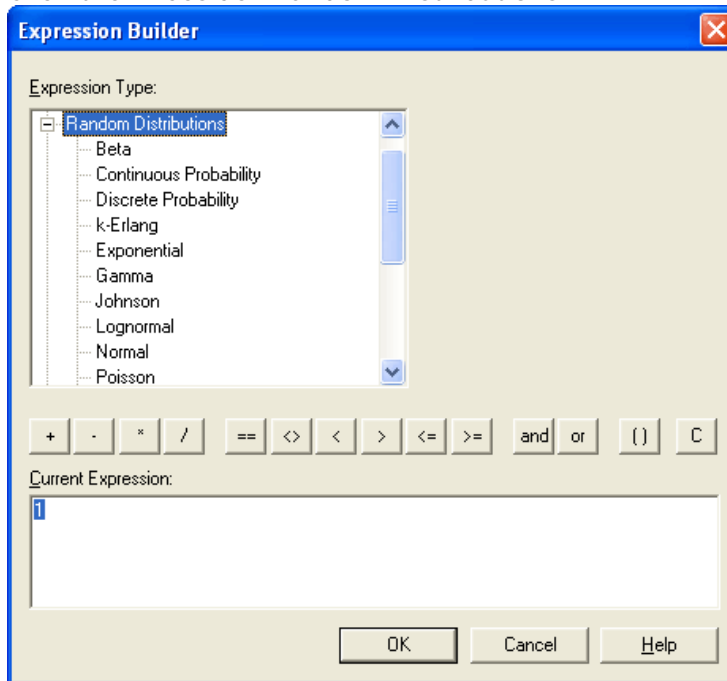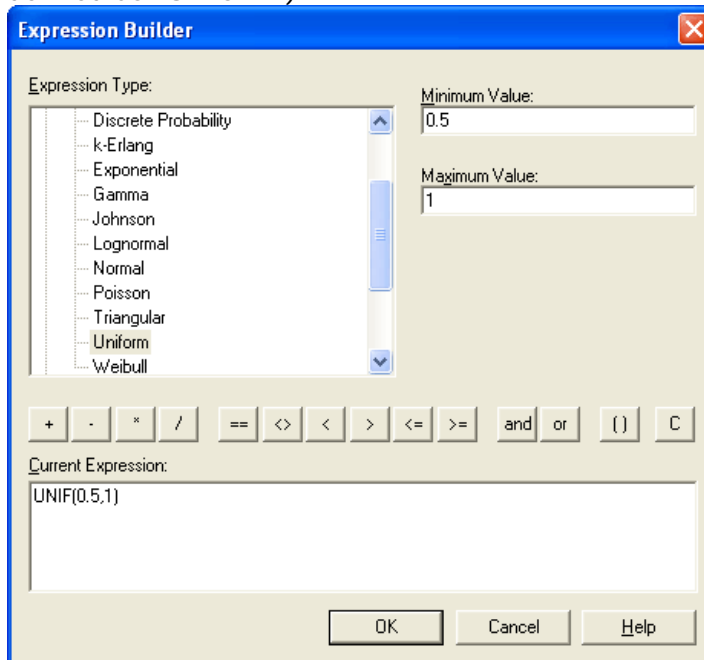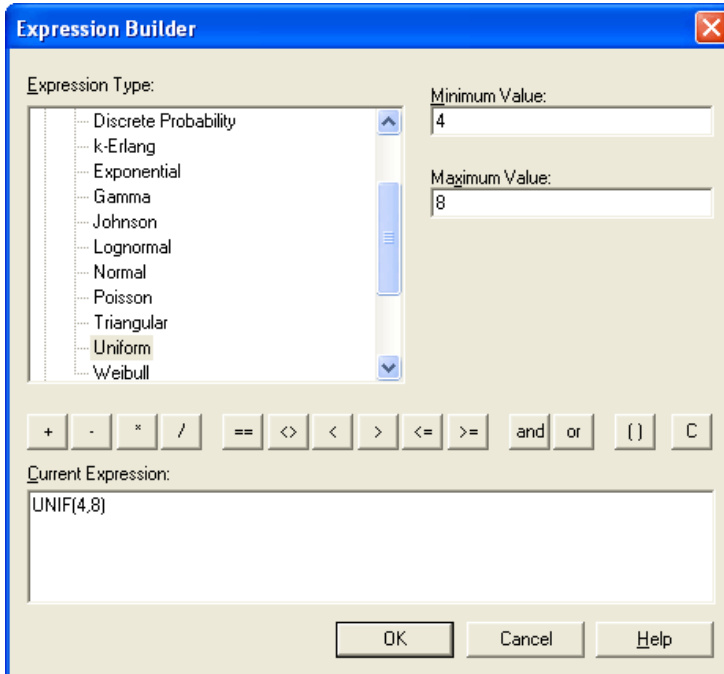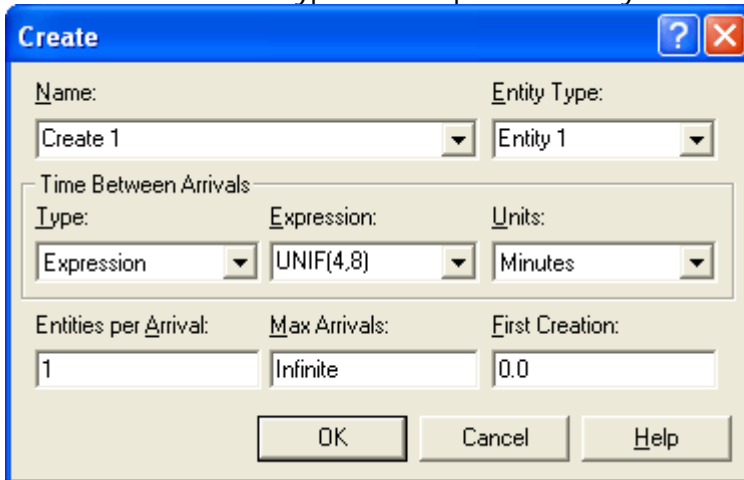

16. Choose the distribution in question (in this case, a Continuous Uniform distribution is defined as "Uniform")

17. Specify the key variables (i.e. the minimum and maximum values), in this case, Minimum = 4 and Maximum = 8.

Expression Builder

Expression Type:
- Discrete Probability
- k-Erlang
- Exponential
- Gamma
- Johnson
- Lognormal
- Normal
- Poisson
- Triangular
- Uniform
- Weibull

Minimum Value:
4

Maximum Value:
8

[+] [-] [*] [/]   [==] [<>] [<] [>] [<=] [>=]   [and] [or]   [( )] [C]

Current Expression:
UNIF(4,8)

[OK]   [Cancel]   [Help]

18. Note how it types the expression for you under Current Expression – this is the language Arena uses to define its expressions – we use the Expression Builder if we aren't familiar with how to write it ourselves.
19. Click OK
20. Note that Arena has typed the expression for you in the "Expression" field

Create

Name:
Create 1

Entity Type:
Entity 1

Time Between Arrivals
Type:
Expression

Expression:
UNIF(4,8)

Units:
Minutes

Entities per Arrival:
1

Max Arrivals:
Infinite

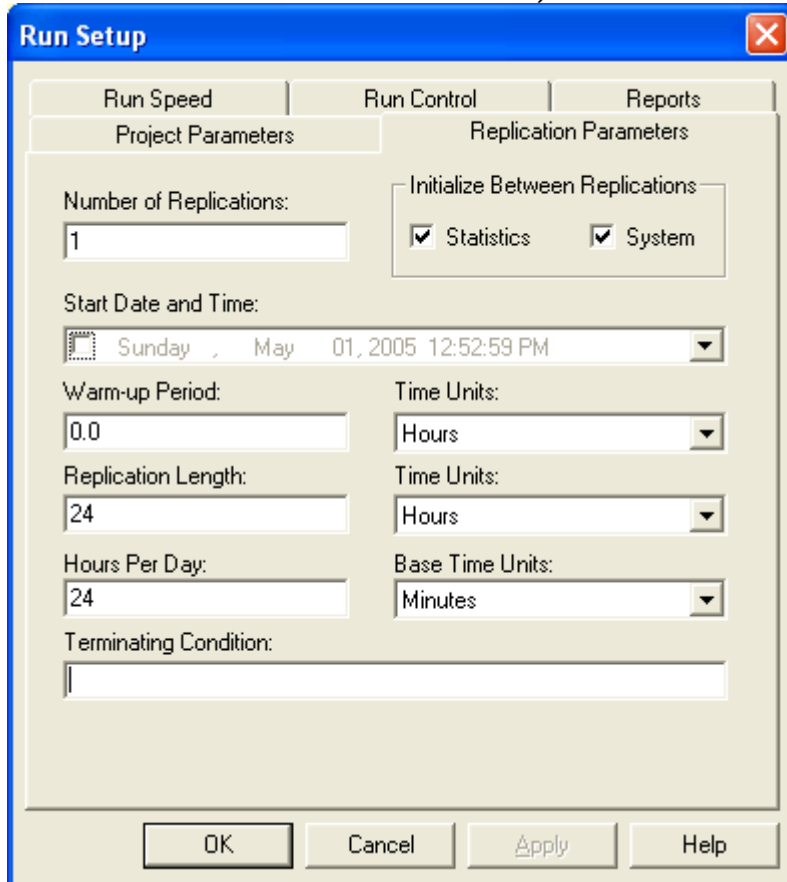First Creation:
0.0

[OK]   [Cancel]   [Help]

21. Click OK

Now that our entities will arrive according to our distribution, we'll set some reporting parameters and analyze the output. We'll attempt to validate our model by analyzing the range of arrival times that Arena uses. They should fall within our distribution.

1. Click Run (at the top)
2. Click Setup

3. Click "Replication Parameters" (at the top)
4. Number of replications: 1 (this is how many times it will repeat the simulation, we just want 1)
5. Replication length: 24 with Time Units: Hours (we'll simulate a whole day, however for more accurate results, you'd normally want 1000 hours)
6. Base time units: Minutes (we want our reporting to be in minutes, since our distribution units are in minutes as well)

7. Click OK

Let's run our simulation now and analyze the reports.
1. Click the "GO" button to begin the simulation.

2. After waiting a little while for the simulation to finish, you'll be asked if you want to see the results, click YES
3. We will cover analyzing the reports in much greater detail later, but quickly analyzing this portion of the report, suggests that our distribution is being simulated correctly:

Other

Number In

	Value
Entity 1	239

This report states that in a 24 hour period (which we defined as the Replication Length), on average (in our 10 replications), 239 entities arrived. Since there are 60*24 = 1440 minutes in a day, and each arrival falls somewhere between 4-8 minutes (according to our Uniform distribution), that means we should expect an **absolute minimum** of 1440/8=180 arrivals per day, **absolute maximum** of 1440/4=360 arrivals per day, and an **average** of 1440/6=240 arrivals per day. Therefore, our model appears valid.

Note: The denominator of 1440/6 is the expected average of the uniform distribution $(b+a)/2 = (8+4)/2 = 6$.

4.4 Discrete Uniform Distributions

A uniform distribution is discrete if all the possible values within the range are countable. For the purposes or inter-arrival times, a discrete uniform distribution would rarely be used; because units of time are continuous (there are so many units that they are uncountable). However, it is worth discussing how to simulate a discrete uniform distribution in Excel, should you need to. **Note:** Often continuous distributions are used to simulate discrete distributions because the number of possible values is countable, but too large.

There are two formulas for the typical types of discrete distributions you'll face.

Formula #1:

This formula will model a distribution of integers between min and max.

Excel formula: =TRUNC(r*((max-min)+1),0) + min

TRUNC:	Excel formula that truncates (removes) the decimal portion of a number to the nth decimal. In this case, n=0.
r:	a reference to a cell that contains a random number between 0 and 1, or the RAND() function could be typed here directly.
max:	The largest number in the distribution, in this case it will be 6
min:	The smallest number in the distribution, in this case it will be 1

To show a real-life use of this formula, we will use the following sample distribution: 1, 2, 3, 4, 5, 6 (min=1, max=6)

In Excel, you would fill in r, max, and min to get:
=TRUNC(RAND()*((6-1)+1),0) + 1

Formula #2:

This formula is more complicated because it can simulate a distribution of real numbers between min and max in 1/f increments, where f >= 1.

Excel formula: =(TRUNC(r*(((max*f)-(min*f))+1),0) + (min*f)) / f

TRUNC:	Excel formula that truncates (removes) the decimal portion of a number to the nth decimal. In this case, n=0.
r:	a reference to a cell that contains a random number between 0 and 1, or the RAND() function could be typed here directly.
max:	The largest number in the distribution, in this case it will be 6
min:	The smallest number in the distribution, in this case it will be 1

f: The denominator of the 1/f ratio. For example, to get 0.5 increments, f would be 2, because 1/f=1/2=0.5. To get 0.25 increments, f would be 4, because 1/f=1/4=0.25.

To show a real-life use of this formula, we will use the following sample distribution:
1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6 (min=1, max=6)

In Excel, you would fill in r, max, min, and f to get:
=(TRUNC(RAND()*(((6*2)-(1*2))+1),0) + (1*2)) / 2

Simulating in Excel:

In order to simplify the input (and to provide an example of this distribution), it has been determined that the time between each arrival happens in 30 second increments between 4 and 8 minutes.

1. You must define which formula we need to use, and what the key variables are. In this case, we need to use 30 second (or 1/2 minute increments), so we need to use formula #2. That makes f=2, min=4, max=8

2. Create an Excel spreadsheet that has the heading "Input" in cell A1 and the heading "Time Between Arrivals (minutes)" in cell A2.

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2					
3					
4					
5					

3. Under the "Input" column, type =RAND() into cell A2

4. Copy/paste/drag this formula down 9 more rows so that you have 10 rows of random input data (in the range 0-1)

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.2194				
3	0.914381				
4	0.25328				
5	0.742612				
6	0.177612				
7	0.301011				
8	0.634975				
9	0.854629				
10	0.583948				
11	0.078362				

5. We must now input our formula filling in r for a reference to the random number, f=2, max=8, min=4

	B2		f_x =(TRUNC(A2*(((8*2)-(4*2))+1),0) + (4*2)) / 2		
	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.446664	6			
3	0.284728				
4	0.403783				
5	0.050714				
6	0.161549				
7	0.664661				
8	0.0029				
9	0.906484				
10	0.009919				
11	0.915361				

6. Copy/paste/drag this formula down to the next 9 rows.

	A	B	C
1	Input	Time Between Arrivals (minutes)	
2	0.717921	7	
3	0.701881	7	
4	0.18704	4.5	
5	0.821445	7.5	
6	0.762683	7	
7	0.045581	4	
8	0.598205	6.5	
9	0.197226	4.5	
10	0.291771	5	
11	0.223101	5	

You now have good estimates as to how many minutes will elapse between each of the next 10 arrivals. As you can see, the values returned fall into the discrete uniform distribution we defined earlier.

Simulating in Arena:

Simulating a discrete distribution in Arena is relatively straightforward. Arena has a "discrete probability distribution" that works very similarly to Excel's VLOOKUP table. If you can formulate a VLOOKUP in Excel, you won't have a problem doing the same in Arena. If you are not yet familiar with VLOOKUP tables, you might want to consider performing the Excel exercise in section *4.9: Custom Distributions* on page 36.

Before we start, we need to define our distribution values, and assign probabilities to each of them. The distribution, like earlier, is 4, 4.5, 5, 5.5, 6, 6.5, 7, 7.5, 8. Since this is a discrete **uniform** distribution, each value has an equal probability. To calculate each value's probability, divide 100% with the number of values in the distribution. In this case, each value's probability is approximately 11% (100/9).

We now need to build a table that we will use as a reference when building our expression in Arena. We need to assign values between 0 and 1 to each of the numbers in our

distribution. We will use the percentage we calculated as the guide for assigning our numbers.

Step 1: List all of the possible values in the distribution.

Step 2: Associate our percentage (11%) in decimal form (0.11) along with the first value in the distribution.

Step 3: Continue assigning numbers between 0 and 1 to the remaining values in the distribution. Each number will be the previous number, plus our given percentage (0.11). Do not fill in the last percentage.
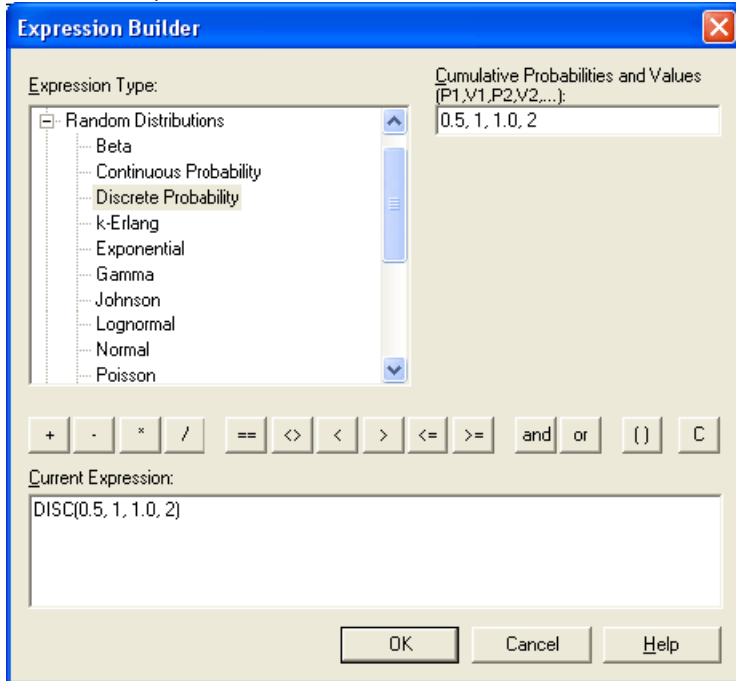
Step 4: Assign 1.0 as the probability for the last value in the distribution. If you notice, it is approximately 0.11 above the last percentage, however, the last percentage must always be 1.0. In reality, the last value in our distribution will appear 1% more frequently than the others, because its probability is 0.01 higher than the others. If we were more precise with our 11% calculation, you could minimize that chance for error. For simplicity, we won't go that far.

4				4	0.11			4	0.11			4	0.11
4.5				4.5				4.5	0.22			4.5	0.22
5				5				5	0.33			5	0.33
5.5				5.5				5.5	0.44			5.5	0.44
6		>>		6		>>		6	0.55	>>		6	0.55
6.5				6.5				6.5	0.66			6.5	0.66
7				7				7	0.77			7	0.77
7.5				7.5				7.5	0.88			7.5	0.88
8				8				8				8	1.0
Step 1				**Step 2**				**Step 3**				**Step 4**	

Now that we've built the table of values and probabilities, we will continue this exercise.
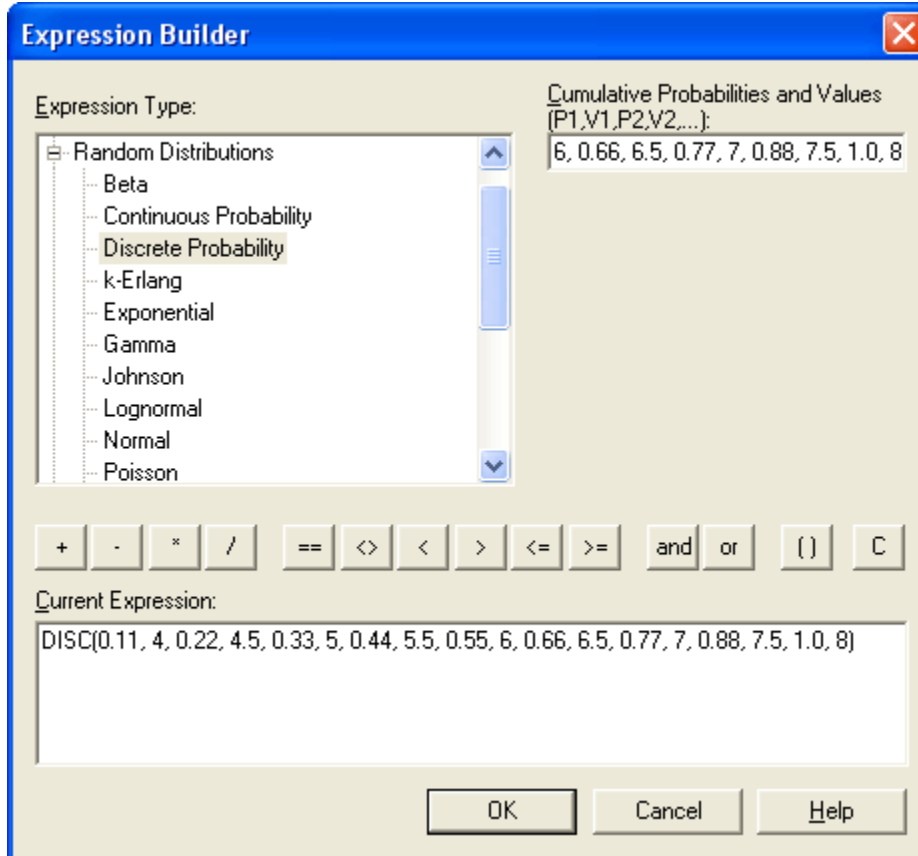
1. Open Arena
2. Save the model (preferably in its own folder)
3. Drag a "Create" module into the work area
4. Click the "Create" module (once) to make sure it is selected
5. Drag a "Dispose" module into the work area
6. Double-click the "Create 1" module that was dragged in
   We can leave "Name" and "Entity Type" as their defaults, as they are only there as labels.
7. Change "Units" to minutes, as our distribution (and variables) are defined in minutes
8. Entities per arrival = 1 (the number of entities that are created)
9. Max arrivals = Infinite (the population limit)
10. First creation = 0.0 (amount of time to elapse before creating the first unit)
11. Under "Type" choose "Expression"
12. Right-click anywhere in the "Expression" dropdown menu
13. Click "Build Expression"
14. Click the + beside "Random Distributions"

15. Choose the distribution in question (in this case, choose the Discrete Probability distribution)
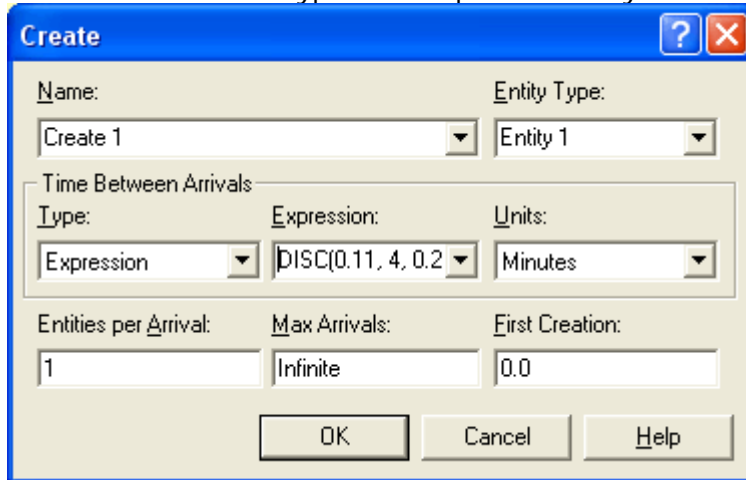

16. Specify the key variables; in this case, it will be a list of probabilities and values. The probabilities are cumulative and need to add up to 1.0. Arena will generate a number between 0 and 1, and will match that number against the probabilities we define. Just like a VLOOKUP table, the probabilities need to be defined as increasing from 0 to 1.0. We have already developed this list at the beginning of this exercise.

In the given field, first write the lowest probability value, then a comma, then its applicable value, then the next probability, a comma, and its applicable value, and so on until all the probabilities/values have been recorded (shown below). **Note:** The last probability should always be 1.0. Do not mix up the placement of the probability and value.

17. Note how it types the expression for you under Current Expression – this is the language Arena uses to define its expressions – we use the Expression Builder if we aren't familiar with how to write it ourselves.
18. Click OK

19. Note that Arena has typed the expression for you in the "Expression" field

20. Click OK

Now that our entities will arrive according to our distribution, we'll set some reporting parameters and analyze the output. We'll attempt to validate our model by analyzing the range of arrival times that Arena uses. They should fall within our distribution.

1. Click Run (at the top)
2. Click Setup
3. Click "Replication Parameters" (at the top)
4. Number of replications: 10 (this is how many times it will repeat the simulation, the more replications, the more accurate the output will be – we'll do 10, but for more important results, you'd want at least 100)
5. Replication length: 24 with Time Units: Hours (we'll simulate a whole day)
6. Base time units: Minutes (we want our reporting to be in minutes, since our distribution units are in minutes as well)
7. Click OK

Let's run our simulation now and analyze the reports.
1. Click the "GO" button to begin the simulation.
2. After waiting a little while for the simulation to finish, you'll be asked if you want to see the results, click YES
3. We will cover analyzing the reports in much greater detail later, but quickly analyzing this portion of the report, suggests that our distribution is being simulated correctly:

Other

Number In

	Value
Entity 1	238

This report states that in a 24 hour period (which we defined as the Replication Length), on average (in our 10 replications), 238 entities arrived. We should validate our model. If we take a weighted average of our inter-arrival times, and calculate how many entities could arrive in a 24 hour time period, we should get a number is close to 238. The weighted average is (11% * 4) + (11% * 4.5) + (11% * 5) + (11% * 5.5) + (11% * 6) + (11% * 6.5) + (11% * 7) + (11% * 7.5) + (11% * 8) = 5.94 minutes between each arrival. Since

there are 60*24 = 1440 minutes in a day, and each inter-arrival time likely falls near 5.94 minutes, that means we should expect an average per day of 1440/5.94=242 arrivals. Therefore, our model appears valid. **Note:** The longer you run the simulation, the closer you'd likely get to the average.

4.5 Normal Distributions

The formula for finding the Z-value (a standard normal random variable) for a normal distribution is as follows:

$$Z = \frac{X - \mu}{\sigma}$$

We want to find out is X (the value) so that we can estimate the inter-arrival times. Shifting this formula around to find X results in:

$$x = Z\sigma + \mu$$

To calculate the Z value of a normal distribution (with mean of 0, standard deviation of 1) and given a probability, we use the following Excel formula:
=NORMSINV(p)
where p is a probability between 0-1

Variables:

μ is the mean of the normal distribution

σ is the standard deviation of the normal distribution

Z is the standard normal random variable (calculated from p)

p is the probability that a variable falls between 0 and values of Z

Example:

μ = 3

σ = 1.2

p = 0.90

Z = 1.28 - calculated using =NORMSINV(0.90)

$$x = Z\sigma + \mu$$
$$x = (1.28 \times 1.2) + 3$$
$$x = 4.54$$

Simulating in Excel:

Following the above example, let's assume that customers arrive at Joe's Pizza Shack with a mean of 3 minutes with a standard deviation of 1.2 in a normal distribution. We want to simulate the time that will elapse between each arrival. **Note:** μ =3, σ =1.2, p and Z will be calculated.

1. Create an Excel spreadsheet that has the heading "Input" in cell A1, the heading "Z Value" in cell A2, and "Time Between Arrivals (minutes)" in cell A3.

	A	B	C	D	E
1	Input	Z Value	Time Between Arrivals (minutes)		
2					
3					
4					

2. Fill 10 rows under "Input" with the RAND() function (this is the p value)

	A	B	C	D	E
1	Input	Z Value	Time Between Arrivals (minutes)		
2	0.277573				
3	0.076301				
4	0.5948				
5	0.208469				
6	0.245731				
7	0.185298				
8	0.378359				
9	0.474031				
10	0.425102				
11	0.328862				

3. Type the formula for finding a Z Value based on a p value in cell B2, refer to A2 for the p value

B2			f_x =NORMSINV(A2)			
	A	B	C	D	E	F
1	Input	Z Value	Time Between Arrivals (minutes)			
2	0.300263	-0.52364				
3	0.914675					
4	0.824533					
5	0.85014					
6	0.611391					
7	0.495241					
8	0.645033					
9	0.690927					
10	0.167419					
11	0.449096					

4. Copy/paste/drag the formula down to the next 9 rows

	A	B	C	D	E
1	Input	Z Value	Time Between Arrivals (minutes)		
2	0.835649	0.976733			
3	0.058757	-1.56529			
4	0.873168	1.141495			
5	0.730713	0.614972			
6	0.869387	1.1235			
7	0.229298	-0.74116			
8	0.015577	-2.1551			
9	0.327517	-0.44678			
10	0.838149	0.98688			
11	0.992908	2.452548			

5. Type the formula for finding X based on the mean, standard deviation, and Z value of a normal distribution in cell C2, refer to B2 for the Z value.

	C2	▼	f_x =(B2*1.2)+3			
	A	B	C	D	E	F
1	Input	Z Value	Time Between Arrivals (minutes)			
2	0.97498	1.959615	5.351538			
3	0.969282	1.870343				
4	0.051068	-1.63459				
5	0.884157	1.196025				
6	0.888064	1.216297				
7	0.342122	-0.40668				
8	0.137209	-1.09295				
9	0.904617	1.308318				
10	0.392866	-0.27186				
11	0.095903	-1.30526				

6. Copy/paste/drag this formula down to the next 9 rows

	A	B	C	D	E
1	Input	Z Value	Time Between Arrivals (minutes)		
2	0.554102	0.136032	3.163238		
3	0.176589	-0.92844	1.885869		
4	0.385492	-0.29109	2.650693		
5	0.131506	-1.1193	1.65684		
6	0.785527	0.790995	3.949194		
7	0.235039	-0.72235	2.133179		
8	0.753987	0.687089	3.824507		
9	0.341673	-0.4079	2.510518		
10	0.097198	-1.29768	1.442778		
11	0.294034	-0.54164	2.350033		

You now have good estimates as to how many minutes will elapse between each of the next 10 arrivals. Should the mean or standard deviation change (or you want to apply this to a different example), simply modify the formula in column C to contain the new values.

Advanced Note:

The Z-Value column could be skipped by using the formula NORMINV() to determine the Time Between Arrivals. It will calculate the value x given the probability, mean, and standard deviation of the normal distribution.

The NORMINV() formula is used as follows:
=NORMINV(p, mean, standard deviation)

In the above example, this function could have been used in cell C2 as:
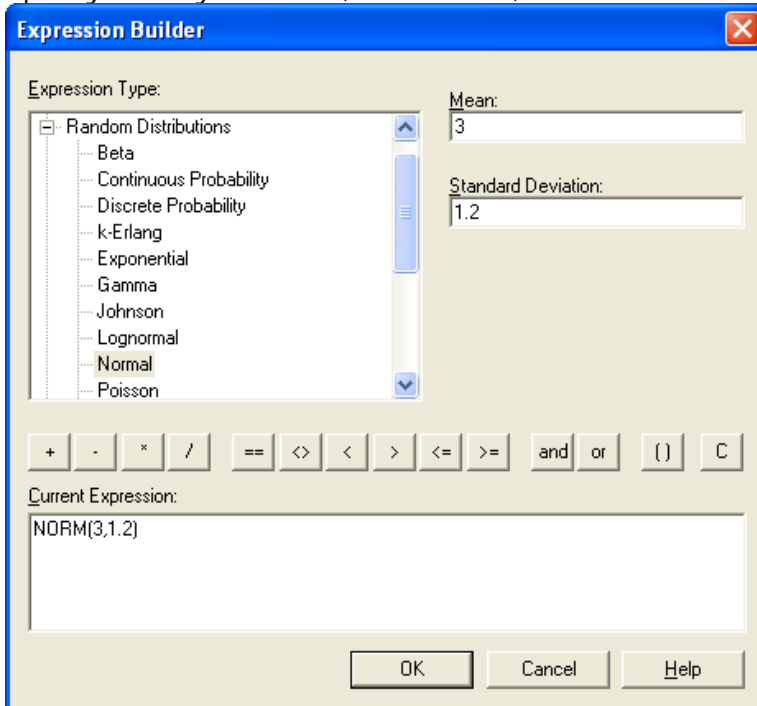=NORMINV(A2,3,1.2)

Using this formula, column B would not be needed.

Simulating in Arena:

1. Open Arena
2. Save the model (preferably in its own folder)
3. Drag a "Create" module into the work area
4. Click the "Create" module (once) to make sure it is selected
5. Drag a "Dispose" module into the work area
6. Double-click the "Create 1" module that was dragged in
   We can leave "Name" and "Entity Type" as their defaults, as they are only there as labels.
7. Change "Units" to minutes, as our distribution (and variables) are defined in minutes
8. Entities per arrival = 1 (the number of entities that are created)
9. Max arrivals = Infinite (the population limit)
10. First creation = 0.0 (amount of time to elapse before creating the first unit)
11. Under "Type" choose "Expression"
12. Right-click anywhere in the "Expression" dropdown menu
13. Click "Build Expression"
14. Click the + beside "Random Distributions"
15. Choose the distribution in question (in this case, a Normal distribution is defined as "Normal")
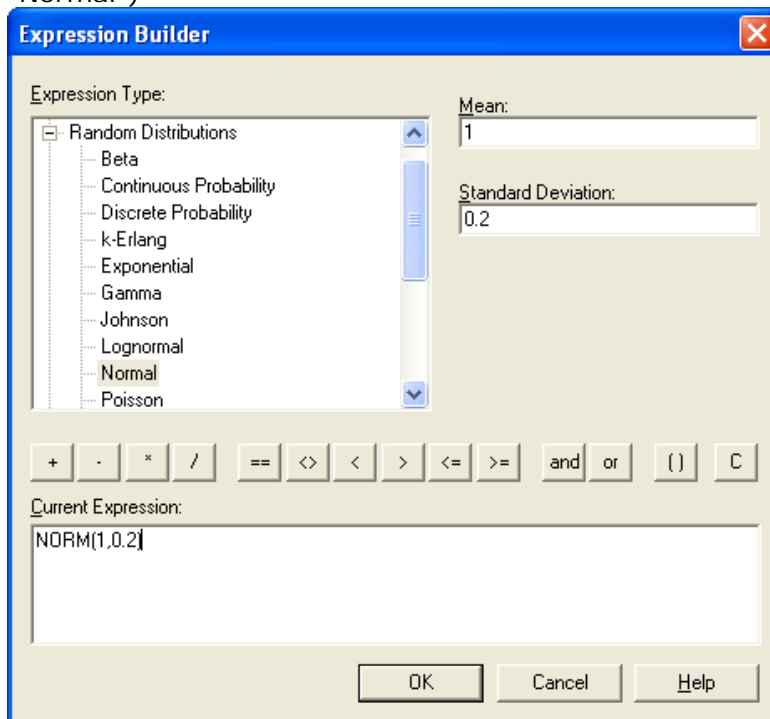
16. Specify the key variables, in this case, Mean = 3 and Standard Deviation = 1.2.

17. Note how it types the expression for you under Current Expression – this is the language Arena uses to define its expressions – we use the Expression Builder if we aren't familiar with how to write it ourselves.
18. Click OK
19. Note that Arena has typed the expression for you in the "Expression" field

20. Click OK

Now that our entities will arrive according to our distribution, we'll set some reporting parameters and analyze the output. We'll attempt to validate our model by analyzing the range of arrival times that Arena uses. They should fall within our distribution.

1. Click Run (at the top)
2. Click Setup

3. Click "Replication Parameters" (at the top)
4. Number of replications: 10 (this is how many times it will repeat the simulation, the more replications, the more accurate the output will be – we'll do 10, but for more important results, you'd want at least 100)
5. Replication length: 24 with Time Units: Hours (we'll simulate a whole day)
6. Base time units: Minutes (we want our reporting to be in minutes, since our distribution units are in minutes as well)
7. Click OK

Let's run our simulation now and analyze the reports.
1. Click the "GO" button to begin the simulation.
2. After waiting a little while for the simulation to finish, you'll be asked if you want to see the results, click YES
3. We will cover analyzing the reports in much greater detail later, but quickly analyzing this portion of the report, suggests that our distribution is being simulated correctly:

Other

Number In	Value
Entity 1	480

This report states that in a 24 hour period (which we defined as the Replication Length), on average (in our 10 replications), 480 entities arrived. Since there are 60*24 = 1440 minutes in a day, and each arrival likely falls near the mean of 3 minutes (according to our Normal distribution), that means we should expect an average per day of 1440/3=480 arrivals. Therefore, our model appears valid.

4.6 Triangular Distributions

The triangular distribution is one of the most widely used distributions in simulation analysis. It is often the distribution of choice when little is known about the probability distribution. It has three parameters: the minimum, the maximum, and the mode. The reason it is used so often is that it is fairly easy to obtain close approximations of the min, max, and mode. For example, you could ask a manager: "What is the absolute minimum time you'd expect between arrivals", "What is the absolute maximum time you'd expect between arrivals?" and "On average, how much time elapses between arrivals?" to get the answers for these three parameters. Of course, data collection would provide more accurate answers (and might even show that a different distribution applies), but the triangular distribution is a good start when you have limited information.

Example:
Min = 4
Max = 12
Mode = 6

Simulating in Excel:

Following the above example, let's assume that customers arrive at Joe's Pizza Shack at a minimum of 4 minutes apart, on average 6 minutes apart, and at maximum 12 minutes apart. We want to simulate the time that will elapse between each arrival.

The formula for computing a Triangular Distribution in Excel is quite complicated. We won't go very far into the theory behind the formula; however, you are encouraged to explore it in greater detail.

Note: The formula makes use of the IF statement in Excel. The IF statement is used as follows:
=IF(expression, what is returned if expr. is true, what is returned if expr. is false)

Note: The formula requires the use of a square root. In Excel, to return the square root of a number, use the following:
SQRT(number)

Formula:
=IF(RN < ((Mode – Min)/(Max – Min)) , LS formula , RS formula)
RN = Reference to a cell containing a random number between 0-1

LS formula = $Min + \sqrt{(RN)(Mode - Min)(Max - Min)}$

RS formula = $Max - \sqrt{(1 - RN)(Max - Mode)(Max - Min)}$

1. Create an Excel spreadsheet that has the heading "Input" in cell A1 and the heading "Time Between Arrivals (minutes)" in cell A2.

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2					
3					
4					
5					

2. Under the "Input" column, type =RAND() into cell A2
3. Copy/paste/drag this formula down 9 more rows so that you have 10 rows of random input data (in the range 0-1)

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.2194				
3	0.914381				
4	0.25328				
5	0.742612				
6	0.177612				
7	0.301011				
8	0.634975				
9	0.854629				
10	0.583948				
11	0.078362				

4. Next, type the full triangular distribution formula (shown earlier) into cell B2, substituting Min for 4, Max for 12, Mode for 6, and RN with a reference to cell A2 (as it will contain the calculated probability for that row).

5. Copy/paste/drag the formula down to the next 9 rows:

You now have good estimates as to how many minutes will elapse between each of the next 10 arrivals. Should the mode, minimum, or maximum change (or you want to apply this to a different example), simply modify the formula in column B to contain the new values.

Simulating in Arena:

1. Open Arena
2. Save the model (preferably in its own folder)
3. Drag a "Create" module into the work area
4. Click the "Create" module (once) to make sure it is selected
5. Drag a "Dispose" module into the work area
6. Double-click the "Create 1" module that was dragged in
   We can leave "Name" and "Entity Type" as their defaults, as they are only there as labels.
7. Change "Units" to minutes, as our distribution (and variables) are defined in minutes
8. Entities per arrival = 1 (the number of entities that are created)
9. Max arrivals = Infinite (the population limit)
10. First creation = 0.0 (amount of time to elapse before creating the first unit)
11. Under "Type" choose "Expression"
12. Right-click anywhere in the "Expression" dropdown menu
13. Click "Build Expression"
14. Click the + beside "Random Distributions"

15. Choose the distribution in question (in this case, a Triangular distribution is defined as "Triangular")
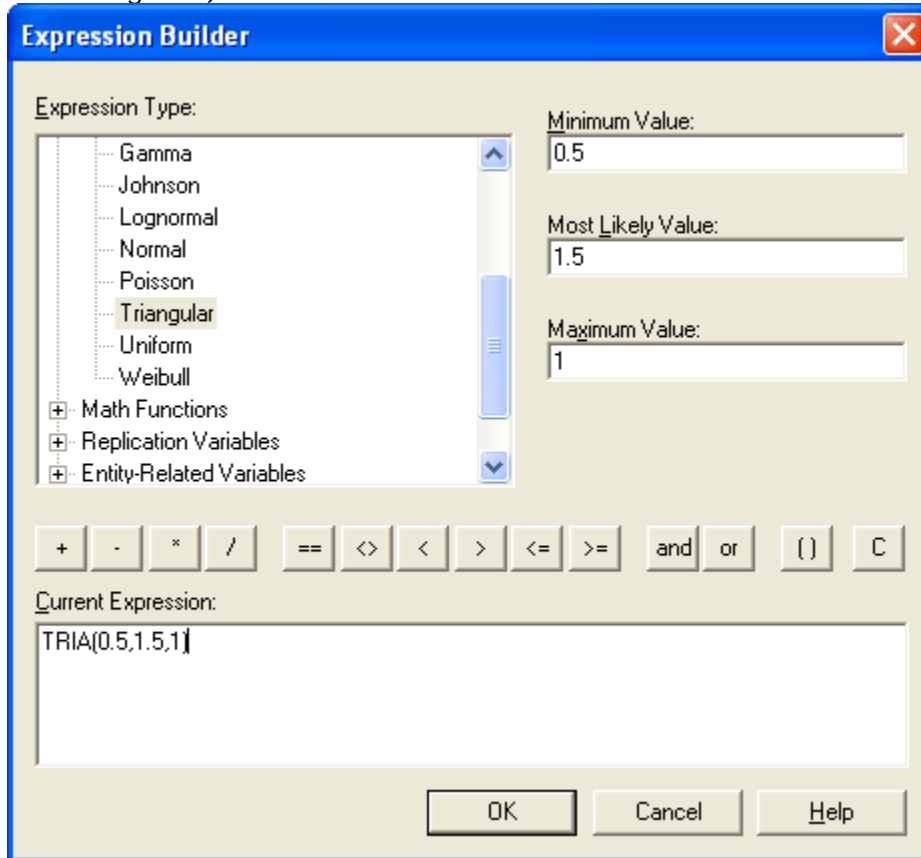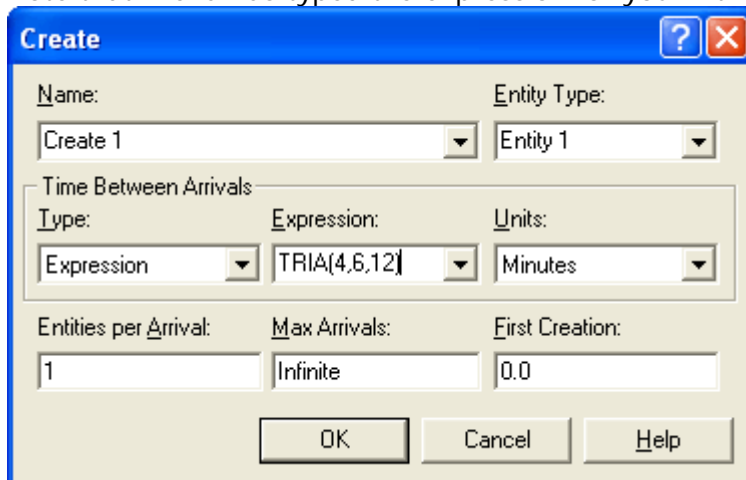
16. Specify the key variables, in this case, Min = 4, Mode=6, Max=12.

17. Note how it types the expression for you under Current Expression – this is the language Arena uses to define its expressions – we use the Expression Builder if we aren't familiar with how to write it ourselves.
18. Click OK
19. Note that Arena has typed the expression for you in the "Expression" field

20. Click OK

Now that our entities will arrive according to our distribution, we'll set some reporting parameters and analyze the output. We'll attempt to validate our model by analyzing the range of arrival times that Arena uses. They should fall within our distribution.

1. Click Run (at the top)
2. Click Setup
3. Click "Replication Parameters" (at the top)
4. Number of replications: 10 (this is how many times it will repeat the simulation, the more replications, the more accurate the output will be – we'll do 10, but for more important results, you'd want at least 100)
5. Replication length: 24 with Time Units: Hours (we'll simulate a whole day)
6. Base time units: Minutes (we want our reporting to be in minutes, since our distribution units are in minutes as well)
7. Click OK

Let's run our simulation now and analyze the reports.
1. Click the "GO" button to begin the simulation.
2. After waiting a little while for the simulation to finish, you'll be asked if you want to see the results, click YES
3. We will cover analyzing the reports in much greater detail later, but quickly analyzing this portion of the report, suggests that our distribution is being simulated correctly:

Other

Number In	Value
Entity 1	192

This report states that in a 24 hour period (which we defined as the Replication Length), on average (in our 10 replications), 192 entities arrived. Since there are 60*24 = 1440 minutes in a day, and each arrival falls somewhere between 4-12 minutes (according to our Triangular distribution), that means we should expect an absolute minimum of 1440/12=120 arrivals per day and an absolute maximum of 1440/4=360 arrivals per day. The mode of 6 minutes only describes the most frequently occurring inter-arrival time (note: it is **not** the average). At first glance, our model appears valid because the value we see falls within the minimum and maximum, however further investigation (using the theory of the triangular distribution) would be needed to completely validate this model.

Note: Refer to "The Triangular Probability Distribution" (WebCT web site) for further information on the theory behind the triangular distribution (if available).

4.7 Exponential Distributions

The Exponential distribution is a one-parameter distribution made up of the parameter λ. The mean and standard deviation of an exponential random variable are equal to each other: $\mu = \sigma = (1 / \lambda)$. Therefore, if you know any one of these three variables, you can simulate this distribution.

To calculate the value X given a probability in an exponential distribution, use the following formula:
x = -μ x LN(p)
where μ is the mean $1 / \lambda$)
where p is the probability

where LN() is the natural logarithm of a number – the LN() formula in Excel

Simulating in Excel:

Following the above example, let's assume that customers arrivals (measured in minutes) at Joe's Pizza Shack are exponentially distributed with λ = 0.20.

1. Calculate μ

$$\mu = \frac{1}{\lambda} = \frac{1}{0.20} = 5$$

2. Create an Excel spreadsheet that has the heading "Input" in cell A1, the heading "Z Value" in cell A2, and "Time Between Arrivals (minutes)" in cell A3.

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2					
3					
4					
5					

3. Fill 10 rows under "Input" with the RAND() function (this is the p value)

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.2194				
3	0.914381				
4	0.25328				
5	0.742612				
6	0.177612				
7	0.301011				
8	0.634975				
9	0.854629				
10	0.583948				
11	0.078362				

4. Type the formula for finding the value of X in cell B2, refer to A2 for the p value, and 5 as the mean

	A	B	C	D	E	F
	B2	▼	f_x =-5*LN(A2)			
	A	**B**	**C**	**D**	**E**	**F**
1	**Input**	**Time Between Arrivals (minutes)**				
2	0.37686	4.879406				
3	0.533168					
4	0.879692					
5	0.050546					
6	0.840719					
7	0.473558					
8	0.9345					
9	0.610224					
10	0.102395					
11	0.095942					
12						
13						

Note: It may be hard to see, but there is a negative (-) sign before the mean

5. Copy/paste/drag the formula down to the next 9 rows

	A	B	C	D	E	F
1	**Input**	**Time Between Arrivals (minutes)**				
2	0.455621	3.930475				
3	0.297872	6.05545				
4	0.436635	4.143294				
5	0.45819	3.902356				
6	0.421845	4.315586				
7	0.554189	2.951249				
8	0.279276	6.377767				
9	0.192472	8.239024				
10	0.701935	1.769576				
11	0.682787	1.907862				

You now have good estimates as to how many minutes will elapse between each of the next 10 arrivals.

Simulating in Arena:

1. Open Arena
2. Save the model (preferably in its own folder)
3. Drag a "Create" module into the work area
4. Click the "Create" module (once) to make sure it is selected
5. Drag a "Dispose" module into the work area
6. Double-click the "Create 1" module that was dragged in
   We can leave "Name" and "Entity Type" as their defaults, as they are only there as labels.
7. Change "Units" to minutes, as our distribution (and variables) are defined in minutes
8. Entities per arrival = 1 (the number of entities that are created)
9. Max arrivals = Infinite (the population limit)
10. First creation = 0.0 (amount of time to elapse before creating the first unit)

11. Under "Type" choose "Expression"
12. Right-click anywhere in the "Expression" dropdown menu
13. Click "Build Expression"
14. Click the + beside "Random Distributions"
15. Choose the distribution in question (in this case, an Exponential distribution is defined as "Exponential")
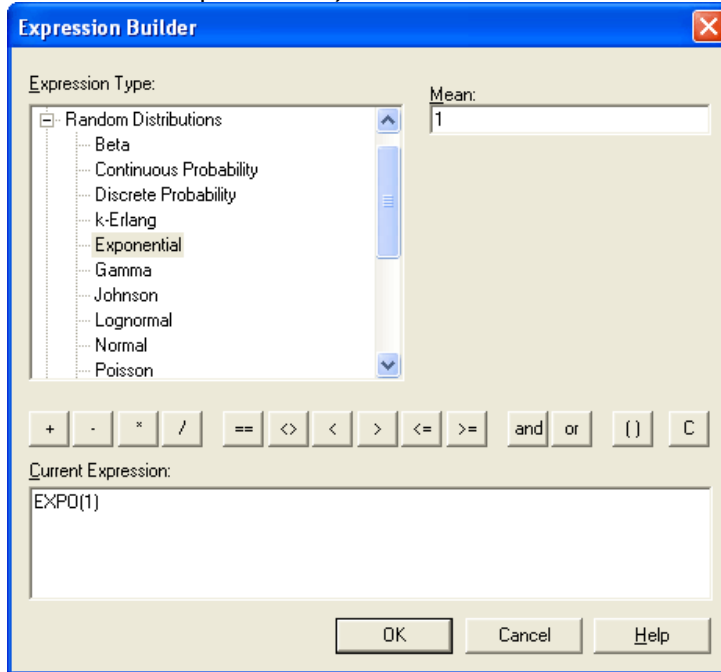
16. Specify the key variables, in this case, Mean = 5 (calculated in Step 1 of the Simulating in Excel section).
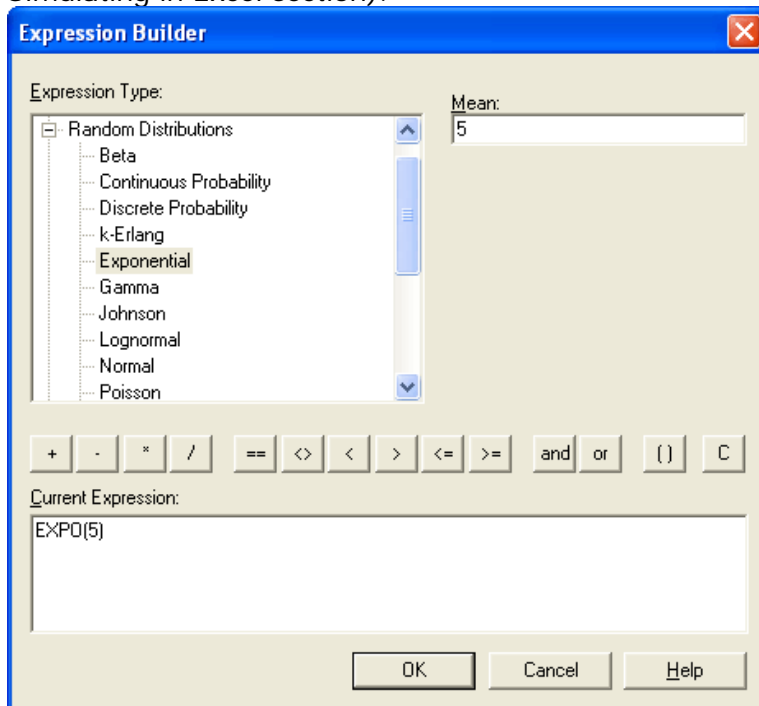
17. Note how it types the expression for you under Current Expression – this is the language Arena uses to define its expressions – we use the Expression Builder if we aren't familiar with how to write it ourselves.
18. Click OK
19. Note that Arena has typed the expression for you in the "Expression" field
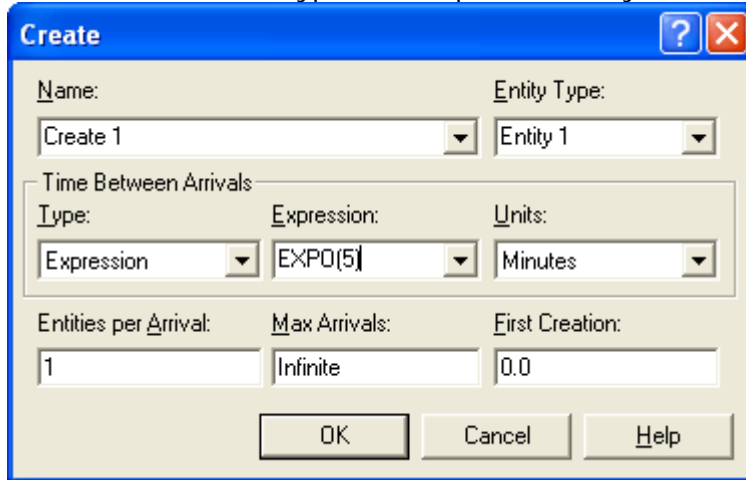
20. Click OK

Advanced Note: Instead of using the expression builder, and because the Exponential distribution is used so frequently, Arena offers it as "Random (Expo)" under the "Type" dropdown. If you select it, you simply have to type the mean (5) next to it (where the whole expression would usually go).

Now that our entities will arrive according to our distribution, we'll set some reporting parameters and analyze the output. We'll attempt to validate our model by analyzing the range of arrival times that Arena uses. They should fall within our distribution.

1. Click Run (at the top)
2. Click Setup
3. Click "Replication Parameters" (at the top)
4. Number of replications: 10 (this is how many times it will repeat the simulation, the more replications, the more accurate the output will be – we'll do 10, but for more important results, you'd want at least 100)
5. Replication length: 24 with Time Units: Hours (we'll simulate a whole day)
6. Base time units: Minutes (we want our reporting to be in minutes, since our distribution units are in minutes as well)
7. Click OK

Let's run our simulation now and analyze the reports.
1. Click the "GO" button to begin the simulation.
2. After waiting a little while for the simulation to finish, you'll be asked if you want to see the results, click YES

3. We will cover analyzing the reports in much greater detail later, but quickly analyzing this portion of the report, suggests that our distribution is being simulated correctly:

Other

Number In	Value
Entity 1	262

This report states that in a 24 hour period (which we defined as the Replication Length), on average (in our 10 replications), 262 entities arrived. Since there are 60*24 = 1440 minutes in a day, and each arrival likely falls near the mean of 5 minutes (according to our Exponential distribution), that means we should expect an average per day of 1440/5=288 arrivals. Therefore, our model appears valid. **Note:** The longer you run the simulation, the closer you'd likely get to the average.

4.8 Poisson Distributions

Poisson distributions are applied when describing the number of entities arriving in a system. A poisson distribution would be described as "the entities arriving are poisson distributed and arrive at a rate of 10 per hour".

In order to simulate this, you must convert 10/hour into the number of minutes (or hours) between each arrival. Since there are 60 minutes per hour, 60/10 = 6 minutes pass on average between each arrival. This is your mean (μ).

When it comes to simulating a poisson distribution, you follow the same **exact same** method you use for simulating exponential distributions using your calculated μ as the mean. So in Excel, the formula remains the same and in Arena, you use the Exponential distribution.

4.9 Custom Distributions

A custom distribution is one where the possible values in the distribution do not fall into any other distribution.

In this case, we know that people arrive to Joe's Pizza Shack in the following manner: 20% of the time 3 minutes will elapse, 50% of the time 5 minutes will elapse, and 30% of the time 6 minutes will elapse between arrivals. We must now assign a range of values from 0 to 1 to each of those three inter-arrival times values so that the selected ranges have the same proportions (20%, 50%, and 30% respectively). The result is the following table:

Inter-arrival Time	Proportion	Range
3 minutes	20%	0.00 – 0.20
5 minutes	50%	0.20 – 0.70
6 minutes	30%	0.70 – 1.00

Note that the numbers that fall between 0.00 – 0.20 are 20% of the possible ranges of values between 0.00 and 1.00. The numbers 0.20 – 0.70 make up 50% of the total possible values, and the range of numbers between 0.70 – 1.00 make up 30%. That is why those particular values were chosen. We must now simulate this distribution in Excel.

Simulating a custom distribution requires the creation of a VLOOKUP table in Microsoft Excel. A VLOOKUP table is a set of ranges, with each range associated with a value. Excel

can take a number (i.e. a probability) and compare it against all the ranges defined in the VLOOKUP table. It will then find the range it falls under, and return the value associated with the applicable range.

The VLOOKUP formula takes the following arguments:

Lookup_Value:	the cell that has the value you are looking up (the probability)
VLOOKUP_Table:	a reference to the entire lookup table
Data_Column:	which column of the lookup table has the return value (i.e. 4)
Range_Lookup:	FALSE if the lookup value must exactly match the range
	TRUE if the lookup value must fall within a range

Simulating in Excel:

1. Create an Excel spreadsheet that has the heading "Input" in cell A1 and the heading "Time Between Arrivals (minutes)" in cell A2.

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2					
3					
4					
5					

2. Under the "Input" column, type =RAND() into cell A2

3. Copy/paste/drag this formula down 9 more rows so that you have 10 rows of random input data (in the range 0-1)

	A	B	C	D	E
1	Input	Time Between Arrivals (minutes)			
2	0.2194				
3	0.914381				
4	0.25328				
5	0.742612				
6	0.177612				
7	0.301011				
8	0.634975				
9	0.854629				
10	0.583948				
11	0.078362				

4. We must now construct our VLOOKUP table, assigning a range of probabilities to each applicable value of the discrete uniform distribution.

5. Construct the VLOOKUP table in columns G-J. You start by typing all the possible values under "Return" – as those are the values you want to "look up". Next, you must assign a range of probabilities to each "return" value. Each range must be equal in its length. We use the range defined earlier, along with the inter-arrival time associated with that range.

G	H	I	J
VLOOKUP			
Range			Return
0	to	0.2	3
0.2	to	0.7	5
0.7	to	1	6

Note: The labels "VLOOKUP", "Range", "Value", and "to" are there for purely visual purposes. They are not required components of a VLOOKUP table. The VLOOKUP table itself is cells G3 to J5. The place in the spreadsheet chosen does not matter.

6. As a matter of exercise, assume you have a random probability value of 0.22. If you look up that value in this table, it falls under the 0.2 to 0.7 range. That means the time between the current and the next arrival will be 5 minutes.

    Advanced Note: It should be noted that the 2nd and 3rd columns of the VLOOKUP table are unnecessary. They are there to make it clearer to the reader what the ranges are. When Excel performs a lookup (i.e. 0.22), it first compares the value with the first range value in column G (i.e. 0). If the lookup value is greater than the range value (i.e. 0.22 > 0) it will continue to the next range value. Since 0.22 > 0.2, it will continue again. Since 0.22 is not > 0.7, it stops. Since it can't go farther than range value 0.2, it has found a match and returns 5.

7. Now, type the VLOOKUP formula in cell B2:

	B2	▼		f_x =VLOOKUP(A2,$G$3:$J$5,4,TRUE)							

	A	B	C	D	E	F	G	H	I	J
1	Input	Time Between Arrivals (minutes)					VLOOKUP			
2	0.285571	5					Range			Return
3	0.128685	3					0	to	0.2	3
4	0.703708	6					0.2	to	0.7	5
5	0.796495	6					0.7	to	1	6
6	0.642177	5								
7	0.424302	5								
8	0.537984	5								
9	0.053869	3								
10	0.294539	5								
11	0.561652	5								

Note: A2 refers to the probability value that will be looked up, $G$3:$J$5 is the reference to the entire lookup table - $ signs are used to denote an absolute reference, 4 refers to the fact that the value being looked up is in the 4th column of the VLOOKUP table, TRUE means that the lookup value can fall within the ranges (i.e. does not have to be an exact match).

8. Copy/paste/drag this formula down to the next 9 rows.

	A	B	C	D	E	F	G	H	I	J
1	Input	Time Between Arrivals (minutes)					VLOOKUP			
2	0.919323	6					Range			Return
3	0.081935	3					0	to	0.2	3
4	0.888724	6					0.2	to	0.7	5
5	0.156787	3					0.7	to	1	6
6	0.897407	6								
7	0.190281	3								
8	0.117775	3								
9	0.14888	3								
10	0.213969	5								
11	0.30733	5								

You now have good estimates as to how many minutes will elapse between each of the next 10 arrivals. As you can see, Excel will look up each individual random probability value and compare it against the lookup table to determine how many minutes will pass between each arrival. Note also that the values returned fall into the custom distribution we defined earlier.

Note: The custom distribution can also return values with decimals. There does not need to be any pattern to the proportions, or the values returned. That is why it is a "custom distribution".

Simulating in Arena:

Simulating a custom distribution in Arena is relatively straightforward. Arena has a "discrete probability distribution" that works very similarly to Excel's VLOOKUP table. If you can formulate a VLOOKUP in Excel using your custom distribution, you won't have a problem doing the same in Arena.

1. Open Arena
2. Save the model (preferably in its own folder)
3. Drag a "Create" module into the work area
4. Click the "Create" module (once) to make sure it is selected
5. Drag a "Dispose" module into the work area
6. Double-click the "Create 1" module that was dragged in
   We can leave "Name" and "Entity Type" as their defaults, as they are only there as labels.
7. Change "Units" to minutes, as our distribution (and variables) are defined in minutes
8. Entities per arrival = 1 (the number of entities that are created)
9. Max arrivals = Infinite (the population limit)
10. First creation = 0.0 (amount of time to elapse before creating the first unit)
11. Under "Type" choose "Expression"
12. Right-click anywhere in the "Expression" dropdown menu
13. Click "Build Expression"
14. Click the + beside "Random Distributions"

15. Choose the distribution in question (in this case, choose the Discrete Probability distribution)

16. Specify the key variables; in this case, it will be a list of probabilities and values. The probabilities are the **upper** values of each range (i.e. the 0.20, 0.70, and 1.0 we defined in our VLOOKUP table) and the values are the return values for each applicable probability. First write the lowest probability value, then a comma, then its applicable value, then the next probability, a comma, and its applicable value, and so on until all the probabilities/values have been recorded. **Note:** The last probability should always be 1.0. Do not mix up the placement of the probability and value.

17. Note how it types the expression for you under Current Expression – this is the language Arena uses to define its expressions – we use the Expression Builder if we aren't familiar with how to write it ourselves.
18. Click OK
19. Note that Arena has typed the expression for you in the "Expression" field

20. Click OK

Now that our entities will arrive according to our distribution, we'll set some reporting parameters and analyze the output. We'll attempt to validate our model by analyzing the range of arrival times that Arena uses. They should fall within our distribution.

1. Click Run (at the top)
2. Click Setup
3. Click "Replication Parameters" (at the top)
4. Number of replications: 10 (this is how many times it will repeat the simulation, the more replications, the more accurate the output will be – we'll do 10, but for more important results, you'd want at least 100)
5. Replication length: 24 with Time Units: Hours (we'll simulate a whole day)
6. Base time units: Minutes (we want our reporting to be in minutes, since our distribution units are in minutes as well)
7. Click OK

Let's run our simulation now and analyze the reports.
1. Click the "GO" button to begin the simulation.
2. After waiting a little while for the simulation to finish, you'll be asked if you want to see the results, click YES
3. We will cover analyzing the reports in much greater detail later, but quickly analyzing this portion of the report, suggests that our distribution is being simulated correctly:

Other

Number In

	Value
Entity 1	292

This report states that in a 24 hour period (which we defined as the Replication Length), on average (in our 10 replications), 292 entities arrived. We should validate our model. If we take a weighted average of our inter-arrival times, and calculate how many entities could

arrive in a 24 hour time period, we should get a number is close to 292. The weighted average is (20%*3) + (50%*5) + (30%*6) = 4.9 minutes between each arrival. Since there are 60*24 = 1440 minutes in a day, and each inter-arrival time likely falls near 4.9 minutes, that means we should expect an average per day of 1440/4.9=294 arrivals. Therefore, our model appears valid. **Note:** The longer you run the simulation, the closer you'd likely get to the average.

5.0   Systems

5.1 Types of Systems

When looking at a place of business (systems) you will find that no two are ever alike. Each have varying arrival rates, service rates, number of processes, number of servers, and line setups.

5.2 Single-channel, single-phase
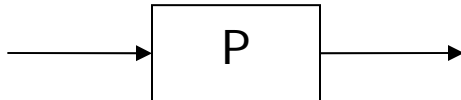
In this system, you have entities arriving into the system, moving through one process, and exiting. All entities form a single line before entering the process. There is one phase (process) and one channel (only one server). **Example:** an ATM machine.

5.3 Single-channel, multi-phase
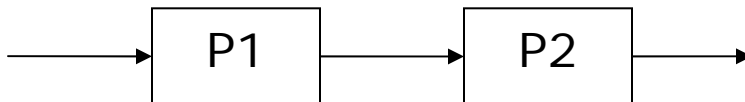
In this system, you have entities arriving into the first process, and as each entity finishes with the first process (P1), they move to the next process (P2). This is a system with sequential processes (you must finish P1 before moving to P2). In this illustration, there is only one server at each process (single-channel) but there is more than one process (multi-phase). **Example:** a car wash (pay first, car washed second).

5.4 Multi-channel, single-phase
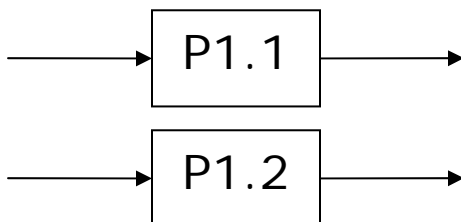
In this system, you have two servers (multi-channel) but only one process (single-phase). In this example, each server has its own line feeding into both processes. **Example:** bank tellers (multiple tellers but only one process: making deposits)

5.5 Multi-channel, multi-phase
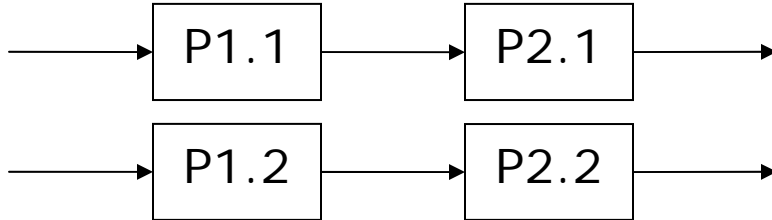
In this system, you have a similar situation but there is more than one process, each having two servers. **Example:** grocery checkout (first you pay, and then your bags are packed).

6.0 Example One

It is the first week of school and students are lining up outside the OneCard office in order to get their OneCard (student identification).

Time between student arrivals is exponentially distributed with a mean of 13 minutes between arrivals. OneCards can usually be created in 8 minutes but sometimes they can be created in as little time as 6 minutes and as long as 12 minutes. Due to illness, only one person will be working the service desk.

6.1 Analysis

Before we begin using Arena, we need to identify the type of system (number of processes, line behaviour, number of servers), distributions (arrivals and service times), and units.

When looking at the OneCard office, you could come up with a few representations of this system. You could conclude that collecting/filling out paperwork is one process, taking your picture is another process, and printing the card is another process. You could of course conclude that there is only one process that contains many tasks (those just mentioned). Here are a few guides to help you in deciding how many processes there are:
- Look to see how many people are working – the more people that are working, generally, the more processes there are likely to be
- Look at the types of tasks people are doing, if there seem to be a series of tasks involved in the system, perhaps each set of tasks warrants a separate process
- Look to see if two (or more) people are doing the **same** set of tasks as each other – if so, that suggests that there are multiple servers for a single process

The more processes you are able to identify, the better your model will be and the better your results will be. However, do not make your model unnecessarily complex. Find the right balance.

In this case, we are going to assume that the entire OneCard creation process is **one process** and since there is only one employee working this process, there is only **one server**. The OneCard office has a **single line** on a **first come first serve basis**. The question states **arrivals in an exponential distribution** and **service times in a triangular distribution**.

Here is a graphical representation of the model:

6.2 Arena

Open Arena:
1. Click Start
2. Click (All) Programs
3. Click Rockwell Software
4. Click Arena 7.0
5. Click Arena 7.01

Save your model:	**Why is this important?**
1. Click File 2. Click Save 3. Choose an appropriate location 4. Choose an appropriate name (i.e. OneCard) 5. Click Save	Saving your model before you begin working on it is important because this establishes the location where your reports will be saved. It is a good idea to save each Arena model you create in its own separate folder, as each model is made up of several files.

Set up the modules:
1. Drag the "Create" module into the main Arena window
2. Making sure the "Create" module is still selected, drag the "Process" module into the main Arena Window
   Note: If you had more than one process in your model, you would continue dragging process modules until you have enough in place.
3. Making sure the "Process" module is still selected, drag the "Dispose" module into the main Arena Window

You should now see this:

If one of your modules did not automatically connect to the previous one, you will need to connect it manually. To do so, click the connect button (shown below):
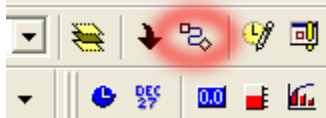
Next, click the blue dot in the first module, and then click the blue dot in the second module (shown below):

Configure the "Create" Module:
1. Double-click the "Create" module
2. Give the module a better name (i.e. "Arrivals")
3. Give the entity type a better name (i.e. "Student")
   Note: An entity is the person/thing that moves through the system. The name of an entity type is merely a label, and does not affect the model.
4. Under "Type" choose the distribution that applies to the inter-arrival rate – we have identified this as exponential, so choose "Random (Expo)"
5. Type the mean of this distribution under "Value"
6. Choose the appropriate units (minutes)
7. On average, only one student arrives at a time so type 1 under "Entities per arrival"
8. The OneCard office has a sufficient population of potential visitors that we can leave "Max Arrivals" as Infinite
9. There is no delay in how soon a student can arrive so leave "First Creation" as zero

10. Click OK

Configure the "Process" Module:
1. Give the module a better name (i.e. "OneCard Creation")
2. Choose "Standard" under Type

Process Types

The different process types in Arena are Standard and Submodel. Standard processing signifies that all logic for this process will be stored within this module. Submodel processing signifies that the logic will be defined in a "submodel" that can include any number of logic (i.e. process) modules. For the purposes of this course, Standard processing is always used.

3. Under "Action" choose "Seize Delay Release" (see the **Process: Actions** box later in this section for more information)

4. We must now add a resource, so click "Add"
5. Give the resource a better name (i.e. "OneCard Employee")
6. Choose the number of employees it takes to move one entity through this process (usually 1).
    Note: This quantity does **not** mean "How many servers are there?" That is set up elsewhere. Do not change this quantity unless it takes a "team" of employees to move **one** entity through the process.

7. Under "Delay Type" choose the appropriate distribution (in this case, it is Triangular)
8. Type in the applicable value(s) for this distribution
9. Choose the units ("minutes")
10. Under "Allocation" choose "Other" (always)

11. Click OK

Process: Actions

Delay: Any entity will be delayed without seizing a resource (i.e. there is no server/resource), meaning that multiple entities can be delayed by this module **at the same time**. This action would be appropriate if, for instance, baked goods are moved into a holding area where they wait for 30 minutes to cool down.

Seize Delay: Any entity entering this process will seize the resource (and therefore prevent any other entity from entering). If using this action, it must be followed by another process module with Delay Release (in order to release the resource that has been seized). For the purposes of this course, this action is not likely to be used.

Seize Delay Release: Any entity entering this process will seize the resource, prevent other entities from entering, and release the resource when finished (allowing other entities to enter). This is the typical action used.

Delay Release: This action is usually used in a module that follows a Seize Delay process module, as it will release a previously seized resource. For the purposes of this course, this action is not likely to be used.

Set up the "Dispose" Module:
1. Double-click the "Dispose" module
2. Give it a better name (i.e. "Happy Students")

3. Click OK
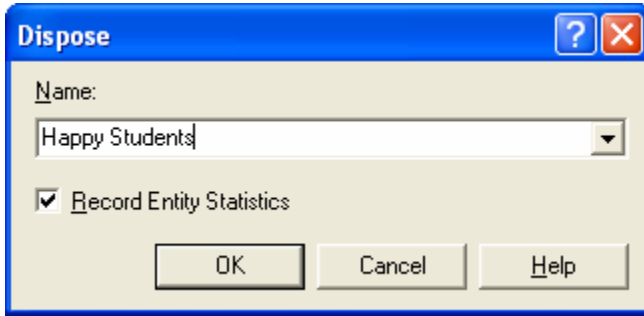
Model:
Your model should now look like this:
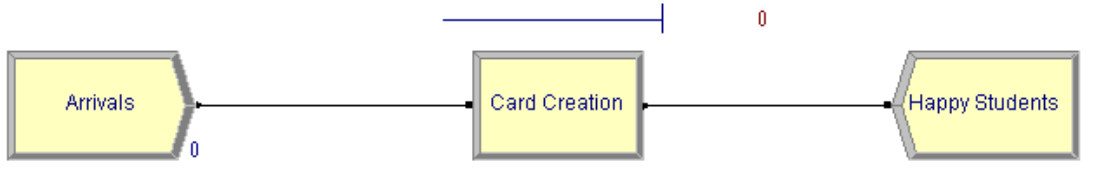
Set number of servers:
1. Click "Resource" in the left menu
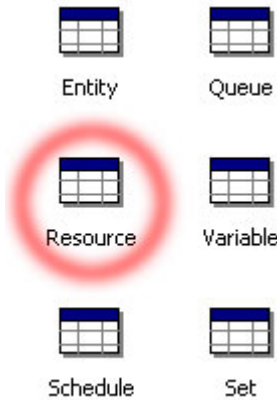
2. This will open the details of all the resources in the system (in the bottom pane)

	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	One Card Employee ▾	Fixed Capacity	1	0.0	0.0	0.0		0 rows	☑

Double-click here to add a new row.

3. To increase the number of servers, just change the "Capacity" to reflect the number of servers.
   Note: You should have only one resource per Process. If there are two processes, there should be two resources. This ensures that there are no errors in the allocation of employees to the various processes.
4. If you were given an hourly wage for this employee, you would enter that wage into the "Busy/Hour" and "Idle/Hour" fields.
   Note: for wages, the number is usually the same between Busy and Idle. However, machines are often more costly to run while busy.
5. All other areas can remain the same

Set up entity:

1. Click "Entity" in the left menu
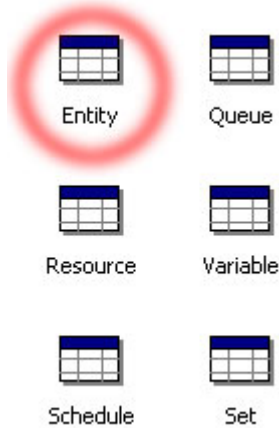
2. This will open the details of all the entities in the system (in the bottom pane)

	Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost	Initial Other Cost	Report Statistics
1	Student	Picture.Report	0.0	0.0	0.0	0.0	0.0	0.0	☑

3. If there were a "waiting cost" given in the question, you would enter that under the "Holding Cost/Hour" column.
4. All other columns will likely remain at zero.

Run Setup:
1. Click Run (at the top)
2. Click Setup
3. Click "Run Speed" (at the top of popup)
4. Adjust the run speed – the higher the number, the faster the animation. If running a simulation for 1000 hours, you should use "100"
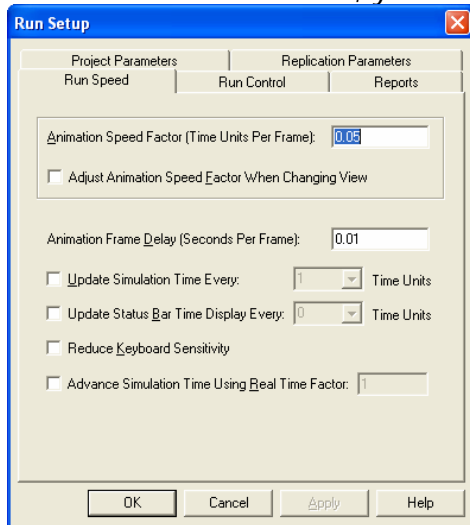
5. Click "Project Parameters" (at the top of popup)

6. Give your project a title and select the reports you would like to see.
   Note: We are not selecting "Costing" because no costs are given in this example.
   You should always select "Processes"
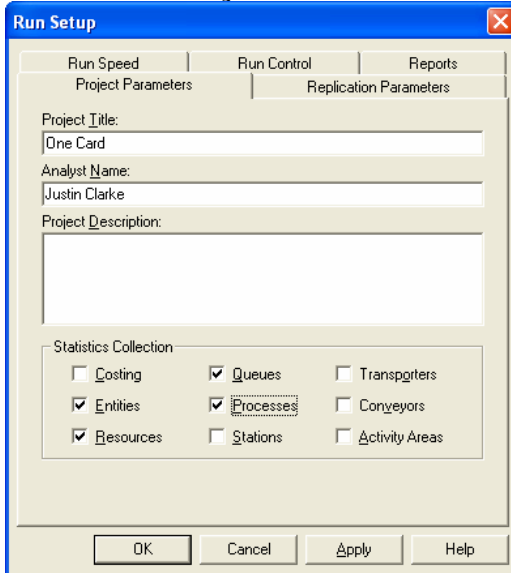
7. Click "Replication Parameters" (at the top of popup)
8. You should run one replication with a length of 1000 hours to get a fairly accurate simulation.
9. Change your "Base Time Units" to the units you wish the report to be in. It is good to keep the base units the same as your arrival and service time units.
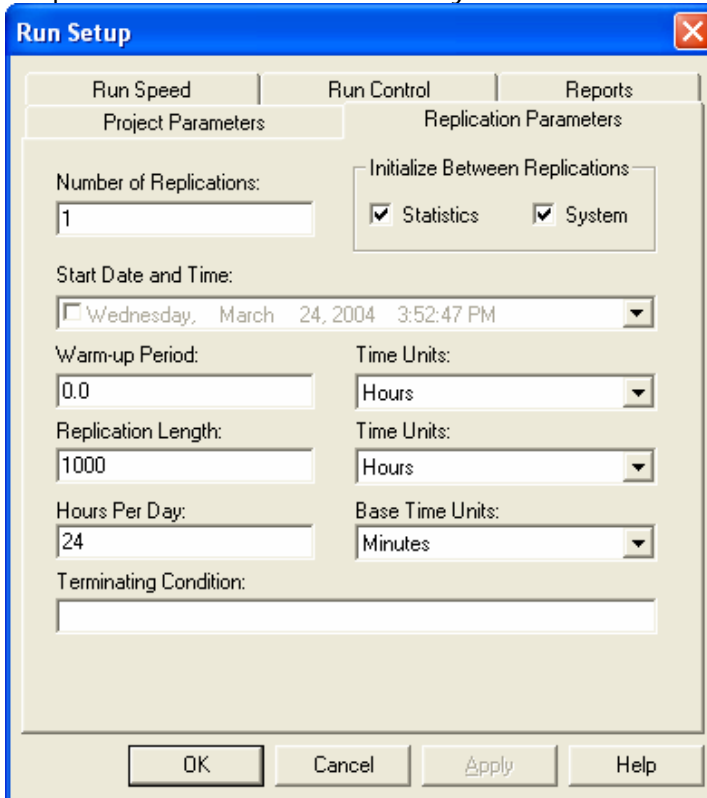
10. Click OK

Run the simulation:
1. Click the "Go" button in the top right corner

2. Watch the simulation
3. Click "Yes" when asked to view the report
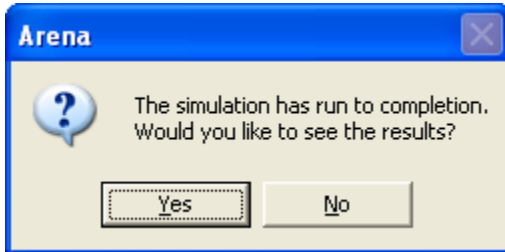
Note: The only way to modify your model after it has been run is to click the END button (five buttons over from Go).

Note: If your simulation crashes, it could be due to a few things:
- Your units in your "Create" module are set incorrectly
- Your units in your "Process" module(s) are set incorrectly
- You have accidentally used a service rate in the "Create" module and an arrival rate in the "Process" module(s) (thus creating an indefinitely growing queue)
- You told the system that 2 (or more) servers are required to complete a process (in the process module) but fewer than that number of servers are actually scheduled to work (the capacity column under "Resource" in the left menu). You likely meant to put a capacity of 2 under "Resource" but put it in the process module instead (which is a mistake).

Analyze Report:

Here is the navigation portion of Arena's reporting program:
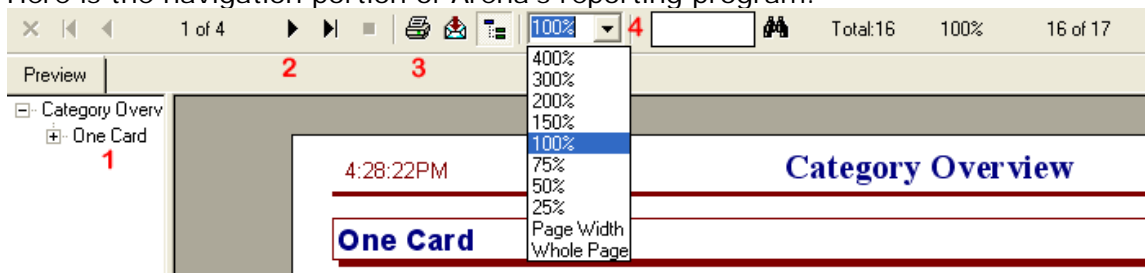
1. Use this to navigate between the various sections of the report
2. Use this to flip to the next page of the report
3. Use this to print the report
4. Use this to choose magnify the report (to zoom in)

We'll use the report to answer a series of questions.

Q: How long does it take for a student to get their new OneCard?

First, analyze the question. What is it asking? The question is not asking how long a student had to wait in line (exclusively), nor is it asking about how long it took to prepare the card (exclusively). This question is asking how long it took the student to move through the entire *system*.

The answer can be found on page 2 (Entity report):

Total Time	Average	Half Width	Minimum Value	Maximum Value
Students	17.7221	1.11869	6.0478	68.0207

A: 17.72 +/- 1.12 minutes

Note: You must include the half-width as part of your answer when giving the average value. The half-width is the 95% confidence interval.

Q: How long does a student have to wait in line, on average?

This question is referring to *waiting* specifically so we need to find the wait time. The answer can be found on page 2 (Entity report):

Wait Time	Average	Half Width	Minimum Value	Maximum Value
Students	9.0479	1.11286	0.00	60.5400

A: 9 +/- 1.1 minutes

Note: there are two ways to get the answer for this question. The other place to get the answer is on page 5 (Queue report):

Waiting Time	Average	Half Width	Minimum Value	Maximum Value
Card Creation.Queue	9.0474	1.11286	0.00	60.5400

The queue report will break down waiting time by *queue* (waiting line) so you can compare wait times between various processes. The wait time identified in the Entity report is only the *total wait time of entity* and does not provide such a breakdown. Since this is a one-process system, these numbers are the same. If there were more than one process, the entity wait time would be approximately equal to the summation of all the queue wait times listed.

Q: How long does it take to have a OneCard prepared, on average?

This question is asking about the service time associated with preparing a OneCard. This answer is on page 2 (Entity report):

Other Time	Average	Half Width	Minimum Value	Maximum Value
Students	8.6742	0.039354163	6.0478	11.9041

A: 8.7 +/- 0.04 minutes

Note: Just like the situation before, this number is the average time of *all processes*. To break down the time by specific processes, you must make sure to choose the "Processes" report (under Run > Setup > Project Parameters). This is reported on page 4 (Process Report).

Other Time Per Entity	Average	Half Width	Minimum Value	Maximum Value
Card Creation	8.6742	0.039354163	6.0478	11.9041

This report breaks wait time down by process. This report will also show wait time leading into each process but that information is available through the Queue Report.

Q: How many students are waiting in line, on average?

This value is reported on page 5 (Queue Report).

Number Waiting	Average	Half Width	Minimum Value	Maximum Value
Card Creation.Queue	0.6992	0.100920237	0.00	7.0000

A: 0.7 +/- 0.1 students

Q: What was the longest amount of time a student had to wait for his/her card?

This question is referring to the entire system time (not just queue waiting time or process waiting time). This can be found in the on page 2 (Entity Report).

Total Time	Average	Half Width	Minimum Value	Maximum Value
Students	17.7221	1.11869	6.0478	68.0207

A: 68 minutes

Q: How busy is the OneCard worker?

This question is referring to utilization. This can be found on page 6 (Resource Report):

Scheduled Utilization	Value
One Card Worker	0.6703

A: The OneCard Worker is busy 67% of the time.

Analysis/Suggestions:

Your analysis of the situation should include answering questions such as the ones above. You need to provide the reader of your report a clear idea of the waiting times, service times, maximum/minimum times, utilization, etc.

You must now analyze the situation and make recommendations for improvement. You must **test** your recommendations to make sure that your recommendations actually improve the situation. A typical cost/benefit analysis given the tested situation should show that it is a good/bad idea.

We are going to analyze this situation, make a recommendation, test it, and decide if that recommendation is good.

Analysis: Some students must wait far too long (one student waited 68 minutes) to get their OneCards. No student should wait longer than 30 minutes.

Recommendation: Add one other server to the process. Is this feasible? Adding another server may crowd the area (and actually increase service time) but we are assuming that the process can handle one extra server.

Test: We save the model as a new file (so as to not mess up the original), click "Resource" in the left menu and increase the capacity of "OneCard Worker" to 2. We run again and view the report.

Total Time	Average	Half Width	Minimum Value	Maximum Value
Students	9.2672	0.104439985	6.0478	24.9865

The report now shows that no student waited longer than 30 minutes.

Good idea? It is now up to you to prove whether the increase in cost (by adding a new worker) is worth meeting this goal of not waiting longer than 30 minutes.

7.0 Example Two

In this example, we will attempt to model Mr. Lube®. We will go to Mr. Lube, identify the type of system they have, collect data, model it, analyze the report, and make recommendations. We will be monitoring how Mr. Lube performs oil changes.

7.1 Analysis

System:
We are assuming that Mr. Lube staffs only one employee and that Mr. Lube only has one process. This process includes meeting with the customer, performing the oil change, and handling payment. The employee, a mechanic, makes $30/hour.

The system:
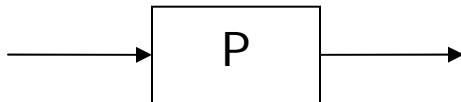
To be more accurate, it would likely be better to separate meeting the customer, performing the oil change, and handling payment into three separate processes. However, for teaching purposes, this is fine.

7.2 Data Collection & Curve Fitting

In any system, you must always collect arrival data. You must then collect data for every single process in the system. During the data collection process, each person should only be responsible for one part of the system. It is far too easy to make mistakes if you need to keep data for multiple parts of the system. Also, if there are multiple servers at a process, each person collecting data should only be in charge of one server at a time.

To collect arrival data, simply write down the time whenever an entity enters the system.

Example:
1:00 PM
1:15 PM
1:50 PM
2:20 PM

To collect service data, simply write down when service begins and when it ends.

Example:
1:00 PM – 1:10 PM
1:15 PM – 1:40 PM
1:50 PM – 2:15 PM
2:20 PM – 3:00 PM

Convert Data:
The next step is to convert the data into the data required by Arena.

For arrival data, you must write down the **inter-arrival** times, meaning, the times between each arrival.

From the arrival data provided earlier, you would end up with:
15   (15 minutes between 1:00 PM – 1:15 PM)
35   (35 minutes between 1:15 PM – 1:50 PM)
30   (30 minutes between 1:50 PM – 2:20 PM)
        (units = minutes)

For service data, you must write down the **service** times, meaning, the time it takes to service the entity.

From the service data provided earlier, you would end up with:
10   (10 minutes between 1:00 PM – 1:10 PM)
25   (25 minutes between 1:15 PM – 1:40 PM)
25   (25 minutes between 1:50 PM – 2:15 PM)
40   (40 minutes between 2:20 PM – 3:00 PM)
        (units = minutes)

These new calculations should be recorded in, for example, an Excel file. These calculations have already been performed for this example. **See Appendix A for the completed calculations.**

Curve Fitting:
It is now necessary to fit this data into a curve and find the expression necessary to tell Arena how to model the data.

1. Write these data values in Excel so it looks something like:

	A	B	C	D
1	**Interarrival Times**		**Oil Change Times**	
2	50		47	
3	33		54	
4	93		56	
5	104		57	
6	71		52	
7	98		45	
8	82		50	
9	98		48	
10	118		54	
11	54		46	
12	112		49	
13	110		50	
14	57		50	

Note: You should have at least 2 hours worth of data and at least 50 data points.

2. Select all of the data associated with Inter-arrival Times (click and drag from cell A2 to AX where X is the last row of data)

	A
1	**Interarrival Times**
2	50
3	33
4	93
5	104
6	71
7	98
8	82
9	98
10	118
11	54

Note: Do not select the header "Inter-arrival Times" – select only data

3. Click Edit
4. Click Copy
5. Open Notepad: Start > (All) Programs > Accessories > Notepad
6. Click Edit
7. Click Paste

You should now see your inter-arrival data in notepad

```
Untitled - Notepad
File  Edit  Format  View  Help
50
33
93
104
71
98
82
98
118
54
112
110
57
107
113
119
60
112
45
43
80
44
65
74
112
```
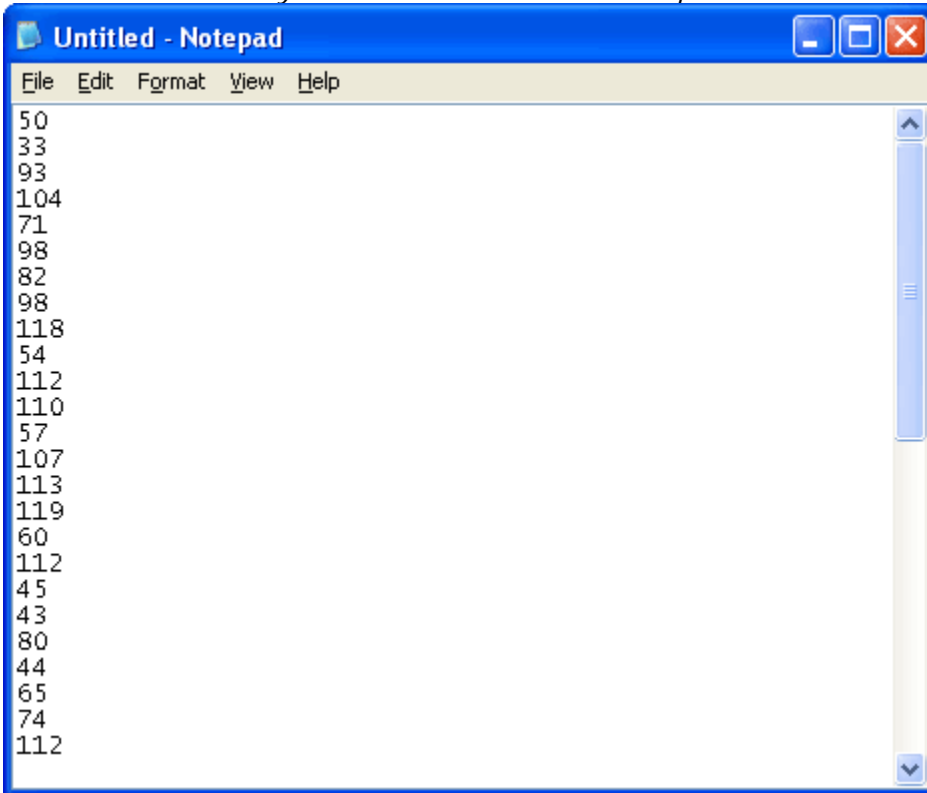
8. Click File
9. Click Save As
10. Choose the folder you've saved your project under

11. Type a good name (i.e. "Arrivals")
12. Click Save

Repeat steps 2-12 for each set of data you've collected. (i.e. repeat the process for the "Oil Change Times" column and save into a file called "Oil Change Times")

The next step is to bring this data into the Arena Input Analyzer and curve fit it.
1. Click Start
2. Click (All) Programs
3. Click Rockwell Software
4. Click Arena 7.0
5. Click Input Analyzer
   (Input Analyzer will now open)
6. Click File
7. Click New
   (You will now have a blank input screen)
8. Click File
9. Click "Data File"
10. Click "Use Existing"
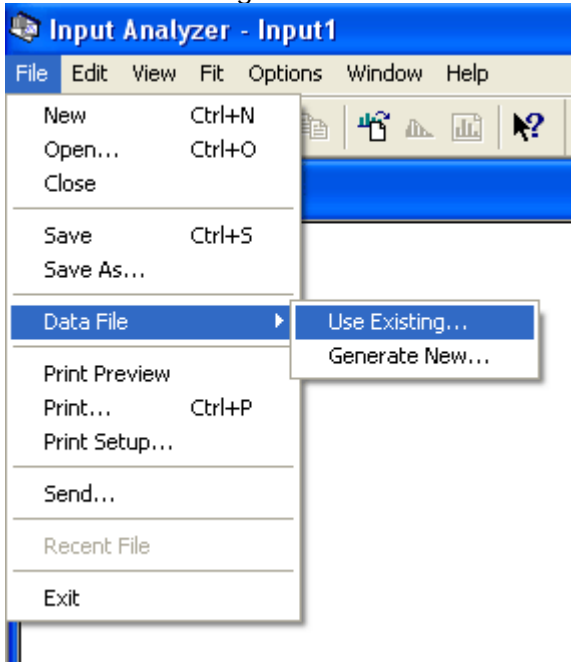
11. Choose the folder where your project is located
12. Beside "Files of type" choose "Text Files"

13. Double-click "Arrivals"
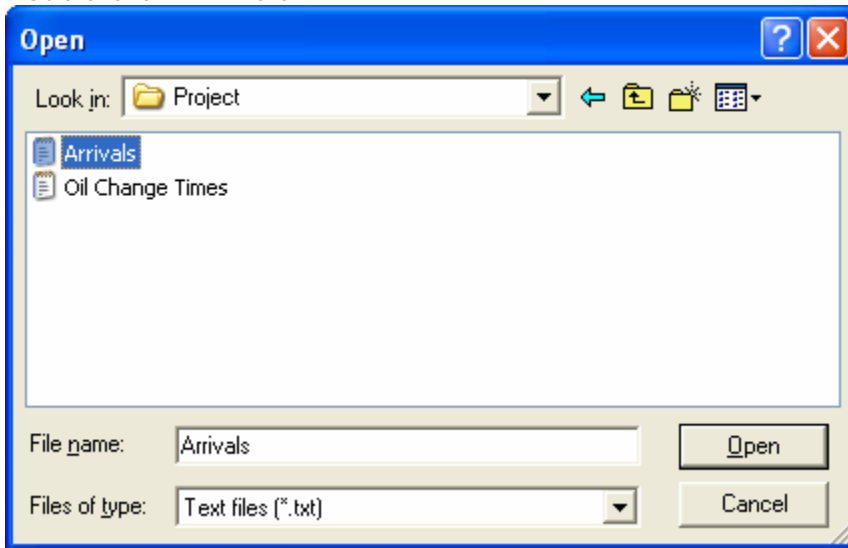
14. You should now see a histogram of all the data.
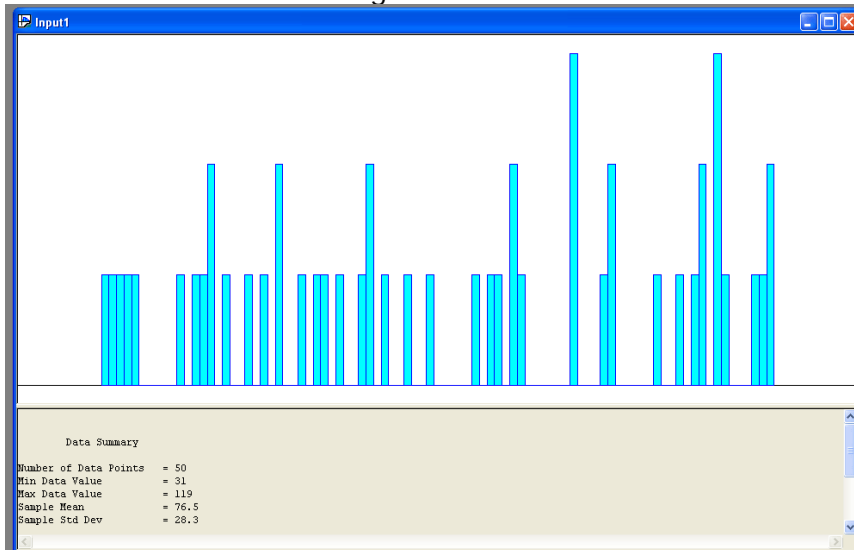
15. Click Fit (at the top)

16. Click Fit All
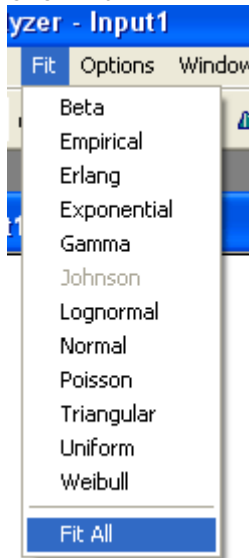
17. You should now see your data with a curve on top
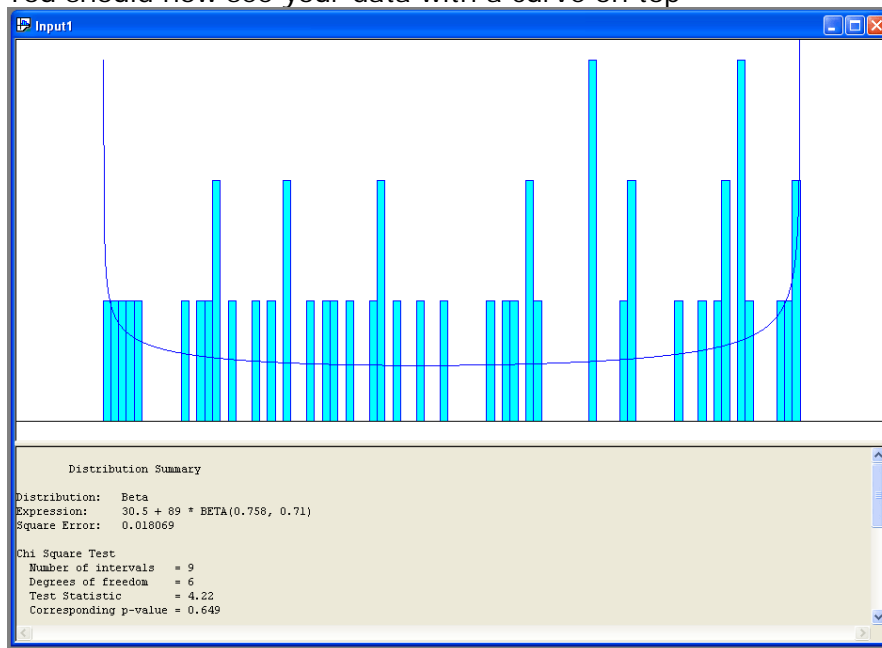
18. Write down the expression that is shown as it is needed later:
    Expression: 30.5 + 89 * BETA(0.758, 0.71)

Repeat steps 6-18 for all your other data. For example, Oil Change Times gives an expression of 44.5 + 14 * BETA(1.06, 1.33)

Once all the expressions are recorded, you can print off these reports if you like (and save them) and then close the input analyzer.

7.3 Arena

The next step is to create the model in arena, input the data expressions, and analyze the report.

We set up the model in Arena just as we did in example one:

Create Module:
1. Double-click the "Create" module
2. Give the module and entity good names
3. Under "Type" choose Expression
4. Type the expression we got from the Input Analyzer for our arrival data:
   30.5 + 89 * BETA(0.758, 0.71)
5. Change the units to minutes
6. Leave all other options the same
7. Click OK

Process Module:
1. Double-click the "Process" module
2. Give it a better name
3. Choose "Seize Delay Release" under action
4. Add a resource (quantity is 1 because it only takes one employee to move an entity through the process)
5. Choose "Expression" under Delay Type
6. Change the units to minutes
7. Choose "Other" under Allocation
8. Type the expression we got from the Input Analyzer for our service data:
   44.5 + 14 * BETA(1.06, 1.33)

9. Click OK

Dispose Module:
1. Double-click the "Dispose" module
2. Give it a better name
3. Click OK

Resources:
We must now properly set up our available resources.
1. Click Resources in the left menu
2. Set the Mechanic's busy/hour and idle/hour rate to 30 each
3. Capacity should be 1 since only one mechanic is working

	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	Mechanic	Fixed Capacity	1	30	30	0.0		0 rows	☑

Run Setup:
1. Click Run
2. Click Setup

3. Set up the various sections to look like this:

4. Click OK

Report:
1. Run the simulation by clicking the GO (play) button
2. Click "Yes" on reading the report

Do the normal analysis of the report pointing out good things and bad things. Two things worth noting is that it cost $30,000 in labour after 1000 hours and that the maximum amount of time a car spent waiting in the system was 116 minutes. After further analysis, we decide that we need to reduce that maximum waiting time to below around an hour.

Alternative #1: Add a server

1. Close the report
2. Click the END (stop) button so we can make further changes
    Note: You should save your project as a different file before making changes to the model.
3. Click "Resource" in the left menu
4. Increase the capacity of the "Mechanic" resource to 2 (thus increasing the number of servers to 2)
5. Run the simulation again
6. Click Yes

The total cost is $60,000 in labour after 1000 hours (huge increase) which drops the maximum wait time to 58 minutes. We have achieved our goal of decreasing the maximum wait time to around an hour, but at a significant increase in costs.

Alternative #2: Split the process into two. The mechanic handles only the tasks that he/she is trained for. All other tasks (customer interaction) are handled by a customer service rep. who is paid at a rate of $8/hour. We estimate that this change will shave 15 minutes off the service time of the current process. We also estimate that the customer service rep. will spend between 6 minutes (minimum), 8 minutes (maximum), and 7 minutes (likely) with the customer.

1. Save your original Arena model into another file
2. Click END (stop) if needed
3. Delete the connection between the Create and Process module (click the connection, hit DEL on the keyboard)

4. Click "Arrivals" so it is selected
5. Drag in another process module between Arrivals and Oil Change
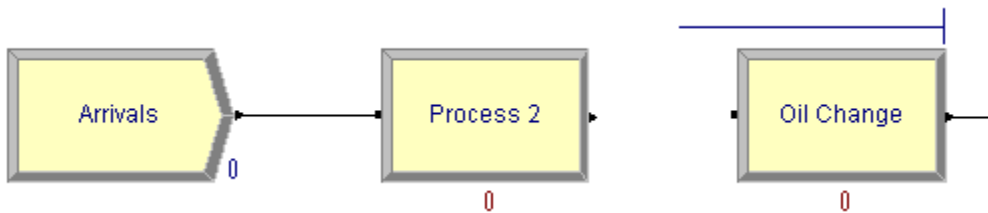
6. Create a connection between "Process 2" and "Oil Change" (click the connection button, click the blue dot of Process 2, click the blue dot of Oil Change)
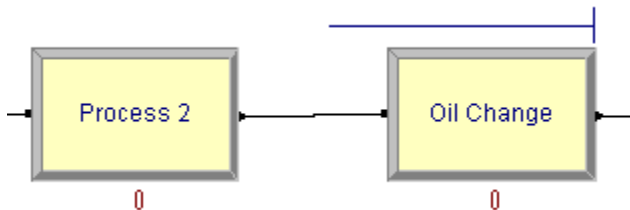
7. Double-click the "Oil Change" module
8. Add "– 15" to the expression in order to subtract 15 minutes from the service time (because we estimated that this service time would be reduced by 15 minutes by moving customer service tasks into the other process)

Expression:

44.5 + 14 * BETA(1.06, 1.33) - 15

9. Click OK
10. Double-click the "Process 2" module

11. Make the necessary changes (give it a name, choose the action, add a resource, set the service time, etc.)

12. Click OK

Set up the resource:
1. Click Resource in the left menu
2. Ensure that the capacity is 1 for both resources (only one of each working at any given time)
3. Change the busy/hour and idle/hour values to 8 for the Customer Service Rep

	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	Mechanic	Fixed Capacity	1	30	30	0.0		0 rows	☑
2	Customer Service Rep	Fixed Capacity	1	8	8	0.0		0 rows	☑

Analyze:
1. Run the report (click GO)
2. Click Yes

We can now note that for an additional $8,000 (total cost $38,000) we can reduce the maximum time to 60.3 minutes.

7.4 Recommendation

Alternative #2 is the best choice. We reduced the total time spent in the system by about 16 minutes at an additional cost of $8,000. Alternative #1 does decrease the total time in the system by an additional 2 minutes; however, it is $22,000 more expensive (that is $11,000 per minute!).

8.0 Advanced: Conditions

Situation: 10% of products coming down a line will likely be rejected due to problems in the production process.

Solution: Use a decision module to redirect 10% of the products elsewhere

1. Create "Create" module
2. Create "Decide" module, make sure it is connected to the "Create" module
3. Double-click the "Decide" module
4. Label it "Reject?"
5. Under "Type", choose "2-way by Chance"
    i. "2-way by Chance" uses a probability to determine TRUE/FALSE
    ii. "2-way by Condition" uses attributes and variables from your model to determine TRUE/FALSE
    iii. "N-way by Chance" allows multiple probabilities to determine TRUE/FALSE
    iv. "N-way by Condition" allows for more than one condition to determine TRUE/FALSE
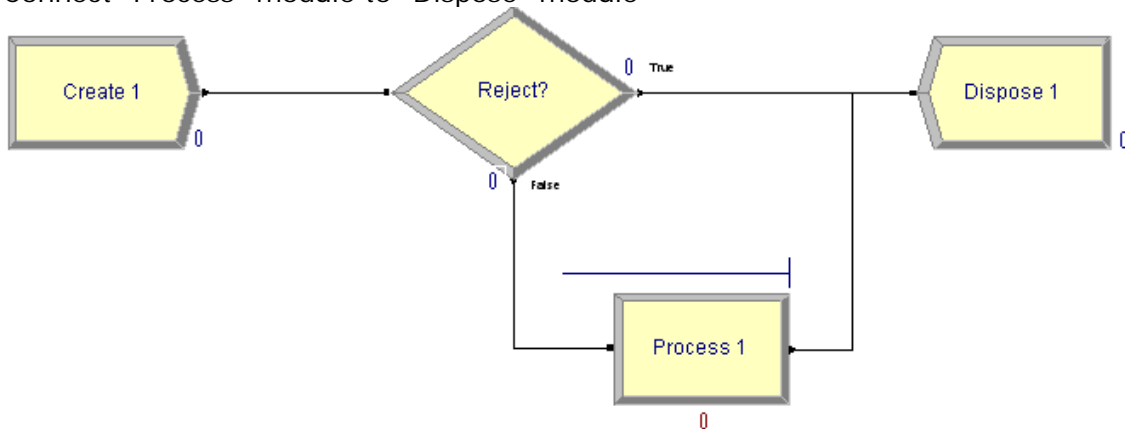6. Type 10% under "Percent True"
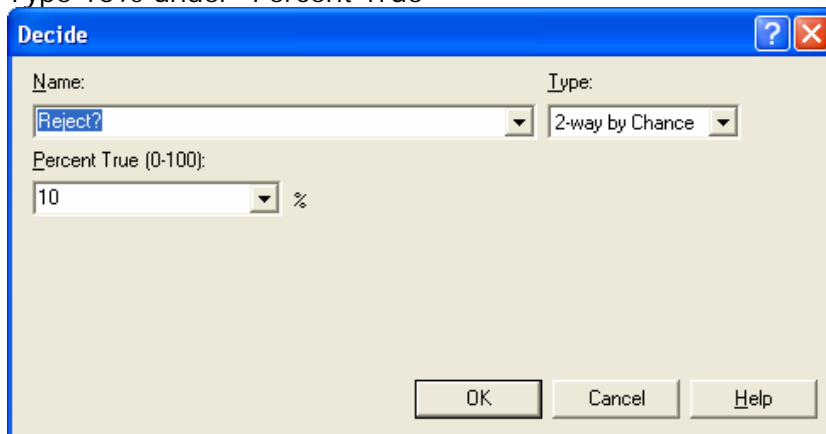
7. Click OK
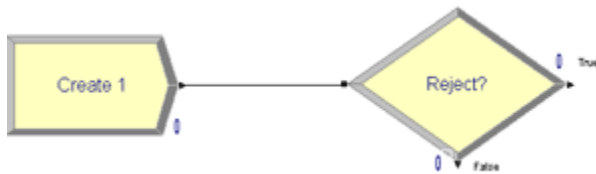
8. Create "Dispose" module
9. Make sure it is connected to the "TRUE" result of the decision module
10. Create "Process" module
11. Make sure it is connected to the "FALSE" result of the decision module

12. Connect "Process" module to "Dispose" module

13. Double-click the "Process" module
14. Choose "Seize Delay Release" under "Action"
15. Add a resource called "Worker" with a quantity of 1
16. Leave all other values as default (the delay type is left as default for this exercise –
    in a real situation, you would set the delay type appropriately)
17. Click OK

18. Zoom to "Page"

Before running this model to see our results, we need to set up the replications, reports, etc.

Run Setup:
1. Click Run (at the top)
2. Click Setup
3. Click the "Run Speed" tab
4. Type "0.01" as Animation Speed Factor
5. Click the "Project Parameters" tab
6. Check "Entities", "Resources", "Queues", "Processes"
7. Click the "Replication Parameters" tab
8. Number of Replications: 10
9. Replication Length: 8 hours
10. Base Time Units: Hours
11. Click OK

Run the model and notice how approximately 10% of entities are being rejected and disposed immediately. All others are processed and disposed like normal.

Discussion:
You could just as easily use a decision module to send an entity to one process over another. For instance, suppose 10% of customers are "preferred" and are allowed to skip a few processes ahead of the other customers. In this case, TRUE might send them to Process #2 where FALSE would send them to Process #1. Decision modules are used heavily in situations where the flow of entities is not always linear from process to process and where skipping processes is allowed.

9.0   Working With / Saving Your Report

Arena generates an .out file that contains the report. If you want to open this report in Microsoft Word, follow these steps:

1. Open Word
2. Click File
3. Click Open
4. Beside "Files of type" choose "All Files"
5. Navigate to the folder where your Arena model is saved
6. Double-click the .out file (it is the name of your arena model followed by .out – ex. Mr Lube.out)
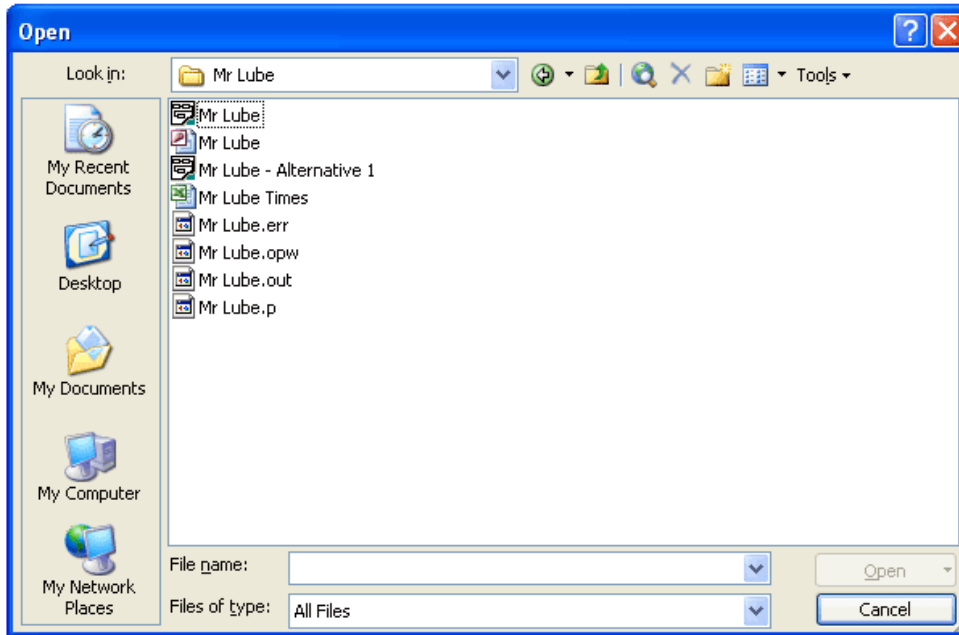
7. The report should look like this in Word:

```
                       ARENA Simulation Results
                         Lab - License: STUDENT


                     Summary for Replication 1 of 1


      Project: Mr Lube                                          Run
      execution date : 3/25/2004
      Analyst: Justin Clarke                                    Model
      revision date: 3/25/2004


      Replication ended at time       : 60000.0 Minutes
      Base Time Units: Minutes


                            TALLY VARIABLES


      Identifier                       Average   Half Width   Minimum
      Maximum    Observations
      _____
      _____

      Oil Change.WaitTimePerEntity         .39339      .19200      .00000
      13.174          783
      Customer Service.OtherCostPerEntity  .93425      .00345      .80710
      1.0576          783
```

10.0 Appendix A: Example Two - Data Points

Inter-arrival Times	Oil Change Times	(units = minutes)
50	47	
33	54	
93	56	
104	57	
71	52	
98	45	
82	50	
98	48	
118	54	
54	46	
112	49	
110	50	
57	50	
107	46	
113	49	
119	56	
60	54	
112	58	
45	46	
43	48	
80	57	
44	53	
65	51	
74	47	
112	45	
109	46	
110	46	
119	52	
97	53	
85	56	
86	49	
66	51	
93	48	
45	47	
85	47	
83	47	
54	53	
117	51	
52	56	
59	49	
31	45	
66	56	
62	50	
68	49	
32	48	
47	51	
41	45	
93	57	
35	54	
34	57	

11.0 Additional Resources

11.1 Arena Manuals

Arena comes with a series of manuals that are very helpful.
1. Open Arena
2. Click Help
3. Click Product Manuals
4. Choose "Arena Basic Edition User's Guide"

11.2 WebCT

The WebCT web site should have a document called "The Triangular Probability Distribution" which is helpful for better understanding the theory behind the triangular distribution.