

***ModelRisk* for Insurance and Finance**

Quick Start Guide



Copyright © 2008 by Vose Software BVBA, All rights reserved.

Vose Software  
Iepenstraat 98  
9000 Gent, Belgium  
[www.vosesoftware.com](http://www.vosesoftware.com)

*ModelRisk* is owned by and a trademark of Vose Software BVBA. ModelRisk product may differ by version. Windows and Excel are registered trademarks of Microsoft Corporation. @RISK is a registered trademark of Palisade Corporation, Ithaca, NY. Crystal Ball is a registered trademark of Oracle Corporation, Redwood Shores, CA. Other brands and their products are trademarks of their respective holders.

## Table of Contents

1	Introduction.....	1
2	Getting Started .....	2
2.1	Installation.....	2
2.2	Start-Up.....	2
3	Overview of <i>ModelRisk</i> .....	4
3.1	ModelRisk Help File.....	4
3.2	Structure of ModelRisk.....	4
3.2.1	Excel Add-In.....	5
3.2.2	Functions vs. Objects.....	6
4	Building Some Models .....	8
4.1	Modeling a Risk Event.....	8
4.1.1	Problem Description .....	8
4.1.2	Conceptual Solution.....	8
4.1.3	Building the Model .....	9
4.1.4	Viewing the Model .....	9
4.1.5	Running the Model .....	11
4.1.6	Conclusion .....	13
4.2	Time Series Fitting and Modeling .....	14
4.2.1	Problem Description .....	14
4.2.2	Conceptual Solution.....	14
4.2.3	Fitting the GBMMR time-series Model to your data.....	15
4.2.4	Running the Model .....	19
4.2.5	Conclusion .....	19
4.3	Simple Aggregate Distribution Model.....	20
4.3.1	Problem Description .....	20
4.3.2	Conceptual Solution.....	20
4.3.3	Building the Model .....	21
4.3.4	Conclusion .....	24
5	Technical Support .....	25

# 1 Introduction

Thank you for trying *ModelRisk for Insurance and Finance* by Vose Software!

We have designed and developed *ModelRisk*, keeping in mind the question, ***‘In a perfect world, what would we, as risk analysts, like to have in a risk analysis software tool?’***

The *ModelRisk* family of products will be a set of tools with each version focused on a particular broad area of risk analysis. The first version *ModelRisk for Insurance and Finance* is a risk analysis tool specifically designed for the insurance and finance industries that allows the user to incorporate advanced techniques in actuarial and financial risk analysis into Excel models. *ModelRisk* gives you access to the most up-to-date, advanced quantitative techniques available without needing to resort to complex programming. With *ModelRisk* you will be able to build very sophisticated and complex spreadsheet-based models in a fraction of the time required to develop custom-coded applications.

*ModelRisk* will be relatively easy to master for anybody familiar with standard Excel functions and interfaces. At the same time, *ModelRisk* provides very sophisticated capabilities for building complex risk models. It may be difficult at first to know where to begin learning how to use *ModelRisk* and start building models. Therefore, this Quick Start Guide is the best place for most new users to begin. It will help you with:

- Getting Started;
- Overview of the software;
- Tutorial on how to accomplish several common tasks with *ModelRisk*;
- References to more information and additional help.

If you are interested in additional help after reviewing this document and working the examples, we have listed a number of additional resources at the conclusion of this Guide.

## **2 Getting Started**

### **2.1 Installation**

To install *ModelRisk* from a CD-ROM, please follow these steps:

1. Insert the *ModelRisk* disc into the CD-ROM drive on your PC.
2. Follow the instructions that appear on the screen.
3. If instructions do not appear, double-click the *ModelRisk* setup.exe file on the CD.

In case you downloaded *ModelRisk* from our website:

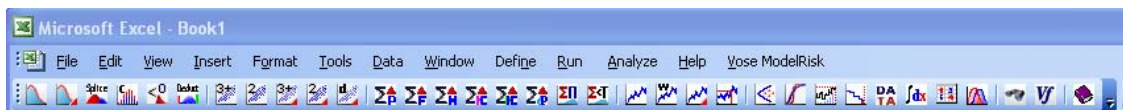
1. Locate the installation file on your hard-drive.
2. Double-click the installation file and follow the instructions that appear on the screen.


If you have any problems or questions, please contact us and we will get back to you as soon as possible (see contact information at the end of this Guide).

### **2.2 Start-Up**

To launch *ModelRisk*, go to Start >> Programs >> Vose Software >> *ModelRisk*.

- You should briefly see a flash screen appear indicating that *ModelRisk* is starting.
- Excel should open automatically:
  - When *ModelRisk* has started there should be a drop-down menu at the top of Excel titled “Vose *ModelRisk*”.
  - You should also see the *ModelRisk* Toolbar.



- The *ModelRisk* icon  will appear in the Windows notification area at the bottom right of your screen.

Possible problems you may encounter:

- If you see the flash screen but the menu and toolbar do not appear within Excel, you may need to select Add-Ins from Excel’s “Tools” drop-down menu and make sure the Vose Functions box is checked.

*ModelRisk for Insurance and Finance Quick Start Guide*

- For Crystal Ball<sup>®</sup> users: the ModelRisk icon needs to be present in the Windows notification area before launching Crystal Ball.
- If you have any other difficulty, please contact us and we will get back to you as soon as possible! You can find our information at the end of this Guide.

## 3 Overview of ModelRisk

### 3.1 ModelRisk Help File

Before using *ModelRisk*, we recommend that you look at the extensive help file provided with *ModelRisk*. To open this file:

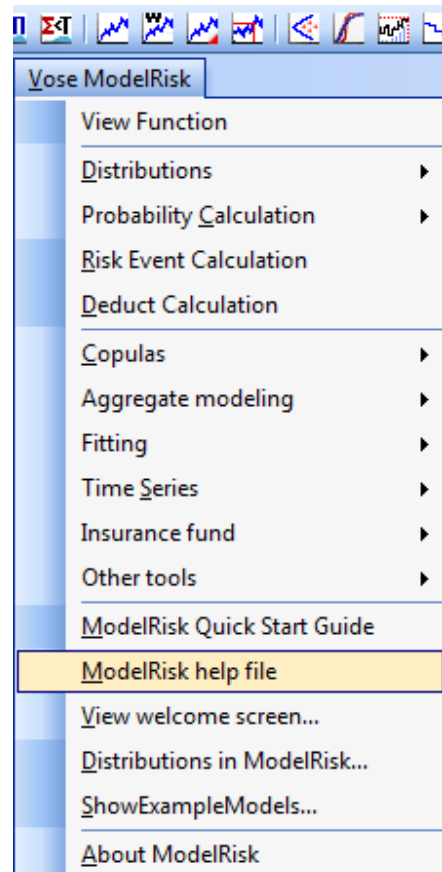
- Go to the “Vose *ModelRisk*” drop-down menu;
- Select the “*ModelRisk* Help File” (see picture below).

In the help file you can find the details of all *ModelRisk* features and how to use them. This help file should be your first reference for any questions you might have regarding *ModelRisk* or risk analysis in general.

Among others, the *ModelRisk* help file includes explanations and information about:

- All built-in probability distributions;
- All available functions;
- The use of objects in *ModelRisk*;
- Illustrations and uses of the many interfaces;
- Background, theory and many practical applications of risk analysis and simulation modeling;
- Topics such as good practices in model building and validation;
- Many of the above subjects are illustrated with example models and videos.

The explanations, descriptions and tools in the *ModelRisk* Help file will help you understand the software quickly, and provide you with a good start to building your own models. The goal of the current “Quick Start Guide” is to give you a brief overview of *ModelRisk*, so that you can quickly become comfortable navigating and using *ModelRisk* and its Help file.



### 3.2 Structure of ModelRisk

Microsoft Excel is used extensively to model business problems and a number of Add-Ins (such as Crystal Ball<sup>®</sup> and @RISK<sup>®</sup>) enable Excel to be used for Monte Carlo simulation and optimization.

However, Monte Carlo simulation Add-Ins provide only the basic building blocks of simulation, allowing one to replace fixed values in a model with random values drawn

from probability distributions. Many sophisticated and highly practical risk analysis techniques have been developed that require more than these basic building blocks. *ModelRisk* makes these techniques available to allow the user to build more complex models and perform sophisticated analysis with ease. Some of these techniques include:

- Stochastic time-series models such as Geometric Brownian Motion, mean reversion, jump diffusion, ARCH, GARCH, EGARCH, continuous time Markov Chain and many others;
- Simulation, statistics and probability calculations for over 70 univariate and multivariate distributions;
- Copulas for modeling complex dependencies;
- Fitting of distributions, time series and copulas to data with facilities to handle missing, truncated and censored data, and options to link to the data for automatic updating and to include uncertainty about parameter estimates;
- Simulation, statistics and probability calculations for aggregate distribution modeling using Monte Carlo, recursive and FFT methods, including aggregate distributions for correlated portfolios;
- Inclusion of retentions and limits in insurance and reinsurance payments at individual and aggregate exposure levels;
- Life insurance modeling;
- Calculations of premiums under several methods;
- Direct calculations of excess loss and VaR at individual or aggregate levels;
- Numerical integration of any bounded Excel/*ModelRisk* equation;
- Extreme value simulation and calculations;
- Portfolio optimization

*ModelRisk* is therefore a companion product to both Excel and any Monte Carlo simulation package for Excel such as @RISK, Crystal Ball or others. With *ModelRisk*, the user can build the sophisticated models listed above, plus many more, in a spreadsheet environment rather than resorting to complex custom coding.

### **3.2.1 Excel Add-In**

*ModelRisk* is a Microsoft Excel add-in that works seamlessly in conjunction with any Monte Carlo simulation risk analysis software. The features in *ModelRisk* can be invoked in exactly the same fashion as regular built-in Excel functions. All of the syntax for the *ModelRisk* functions is explained in detail in the extensive help file.

There are three main ways to insert *ModelRisk* functionality into your model:

1. The *ModelRisk* Toolbar for the most broadly applicable techniques.
2. Many functions are available under the “Vose *ModelRisk*” drop-down menu.
3. All *ModelRisk* functions, objects and distributions can be entered directly in Excel by typing them into the applicable cell or using Excel’s **Insert Function** interface.

Most users will want to build their models using *ModelRisk* directly in a spreadsheet environment. It is also possible to incorporate the functions within *ModelRisk* directly into your own C++, VBA, etc. code by using *ModelRisk*'s developer's kit. For more information, please contact [info@vosesoftware.com](mailto:info@vosesoftware.com).

### 3.2.2 Functions vs. Objects

In addition to providing Monte Carlo functionality, *ModelRisk* offers a unique and very powerful approach to handling probability distributions in risk analysis modeling with the use of '*objects*'.

Other risk analysis software packages only offer the ability to randomly sample from probability distributions to create Monte Carlo scenarios ('iterations'). *ModelRisk* offers this capability (e.g. the function =VoseLognormal(100,10) generates random samples from a Lognormal distribution with a mean of 100 and a standard deviation of 10) but in addition, *ModelRisk* also allows you to define a distribution as an *object* which can then be manipulated in many ways as we will see in the examples below. The use of *objects* has, until now, only been available in high-end mathematical research software or in certain programming languages. *ModelRisk* pioneers the next generation of risk analysis software, bringing the power and flexibility of probability objects to the intuitive, familiar environment of Excel.

For example, in Cell A1 in Excel we could write: "= VoseGammaObject(2,3)" which defines a Gamma(2,3) distribution object. Then writing in a different cell:

- "=VoseMean(A1)" will return the mean of a Gamma(2,3) distribution.
- "=VoseStopSum(A1, 100)" will simulate the number of Gamma(2,3) distributions that would be needed to exceed a total of 100.
- "=VoseAggregateMC(200, A1)" will randomly simulate the sum of 200 Gamma(2,3) distributions.
- "=VoseRiskEvent(0.2,A1)" will generate values of either 0 (with a 0.8 probability) or a random value of the Gamma (2,3) with a 0.2 probability.

Some other examples are:

- "=VoseAggregateFFT(VosePoissonObject(10), VoseLognormalObject(2,5))" will construct the aggregate distribution of Poisson(10) variables following a Lognormal(2,5) distribution using Fast Fourier Transform methods.
- Similarly, "=VoseAggregateFFT(VosePoissonObject(10), VoseLognormalObject(2,5), 0.99)" will return the 99<sup>th</sup> percentile of this aggregate distribution *directly* from calculation rather than simulation.

*ModelRisk* has a large range of functions that can make use of *objects* to solve common risk analysis problems. Some of the advantages of *objects* include:

- Greatly simplified risk analysis models;
- Distribution assumptions made clearer and visible on screen;
- Reduced model size;
- Speeding up simulations by several orders of magnitude; and
- Improved accuracy.

*Objects* help reduce errors in model building by automating much of the complex mathematics leaving you to focus on the problem, instead of fighting with formulae in Excel.

*Objects* can even be created for distributions fitted to data (all linked 'live', so that if you update your data, the full model gets updated). For example:

- “=VoseMean(VoseGammaFitObject(*data*))” will return the mean of a Gamma distribution fit to an array of values '*data*';
- “=VoseMean(VoseGammaFitObject(*data*, TRUE))” will return the statistical uncertainty about this mean;
- “=VoseAggregateFFT(VosePoissonFitObject(*data1*), VoseLognormalFitObject(*Data2*),0.999)” will calculate the 99.9<sup>th</sup> percentile of the aggregate distribution where the frequency and claim size distributions are fit to data sets *data1* and *data2* respectively; and
- “=VoseMean(VoseDeductObject(VoseGammaFitObject(*data*,),10,90, TRUE))” will return the mean of a Gamma distributed cost fit to data with a retention of 10 and a maximum limit of 90.

We are sure that once you start using distribution *objects* in your risk models, you will quickly see how they can make your modeling quicker and much less prone to errors, easier to track and modify, and much simpler to follow.

## 4 Building Some Models

We have provided here several example models that use different features of *ModelRisk*. These are presented in a “tutorial” format and you are encouraged to work through them in detail in order to become familiar with the *ModelRisk* software.

### 4.1 Modeling a Risk Event

#### 4.1.1 Problem Description

Imagine that you are involved in managing a particularly complex project (for example, building a bridge or construction of an airplane). You have identified five risks as being particularly important for finishing the project within budget. If any of these risks occur, there will be a negative financial impact on your budget. You wish to model the total risk associated with this set of five risks. For each risk there is a unique probability of occurrence. If an event occurs, there is a financial impact which follows a specific distribution unique to that event (see table below).

Risk Event	Probability of occurrence	Impact Distribution (US\$1000's)
A	0.2	Lognormal(6,3)
B	0.1	Exponential(3)
C	0.4	Pareto(6,2)
D	0.02	Gamma(3,4)
E	0.5	Weibull(14,2)

Table 1 - List of risks with the probability of occurrence and impact

For example, Risk Event A has a 0.3 probability of occurring and if it does occur, the resulting impact of the risk follows a lognormal distribution with mean of US\$6,000 and standard deviation of US\$3,000.

#### 4.1.2 Conceptual Solution

In this example, we are assuming that all five risks are independent of each other. For each risk event, we will build a model which essentially generates a probability distribution that combines both the risk and its impact. If we then take a random sample from each of these combined distributions and sum the samples, the total will be one iteration (possible scenario) of the many potential results of these risks. Running a Monte Carlo simulation and collecting the resulting sum values will therefore generate a probability distribution of the total financial risk associated with these five risks.

### 4.1.3 Building the Model

VoseRiskEvent() is a *ModelRisk* built-in facility designed specifically for this situation. We can use the function to model this problem in a very straightforward and compact form, as shown in Figure 1. For all our examples we will follow the formatting convention of inputs as blue, outputs as red, and calculations and data labels as black.

	A	B	C	D	E	F
1						
2		Risk Event	Probability	Impact	Event	
3		A	0.2	VoseLognormal(6,3)	2.8978	
4		B	0.1	VoseExpon(3)	0	
5		C	0.4	VosePareto(6,2)	0	
6		D	0.02	VoseGamma(3,4)	0	
7		E	0.5	VoseWeibull(14,2)	1.7611	
8				Sum	4.6589	
9						
10		<b>Formulae table</b>				
11		D3 (D4:D7 similar)		=VoseLognormalObject(6,3)		
12		E3:E7		=VoseRiskEvent(C3,D3,RiskUniform(0,1))		
13		E8		=RiskOutput() + SUM(E3:E7)		
14						

Figure 1 - Risk Event model with ModelRisk

### 4.1.4 Viewing the Model

- As an initial test of your model, you can press the F9 key (Excel's recalculate shortcut). Each time you press the key, a random value is generated from the Risk Event distribution for each risk. Hitting the F9 key a number of times will give you a feel for the behavior of the model.
- Next, for verification purposes, you might want to visualize some of the components of the model. You can do this by selecting a cell and clicking the View Function (V) button on the *ModelRisk* toolbar to display the underlying formula and distribution (you can click the View Function button for almost every feature in *ModelRisk*).
- Try this for both the Risk A Impact cell and Risk Event cell and you will see the two windows shown in Figure 5 and Figure 6:

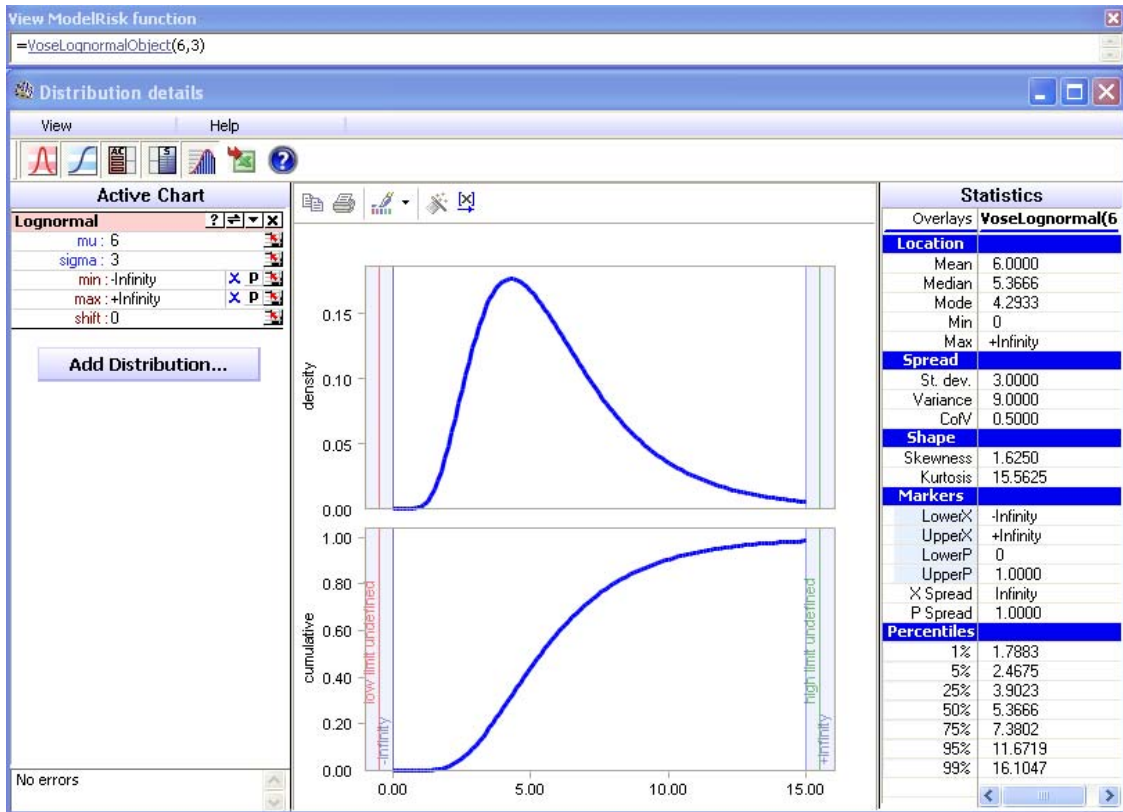


Figure 5 - Formula View of Risk Event A Impact cell (cell D3)

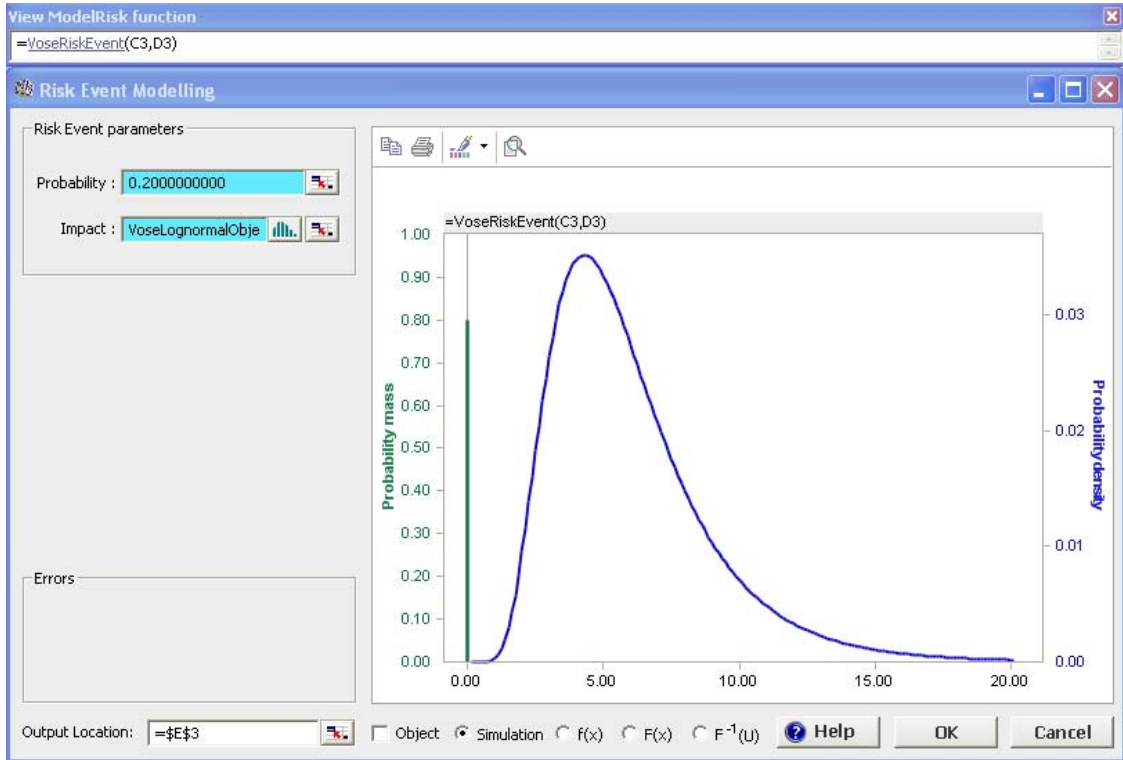


Figure 6 - Formula View of Risk Event A (cell E2)

### 4.1.5 Running the Model

Now it is time to run the model so we can determine the resulting total risk associated with these five project risks. Therefore, follow the next steps:

- Set up your simulation software for at least 5,000 iterations of the model and run the simulation.
- If you generate a histogram with 50 bins from the resulting data, it should look very close to Figure 7 shown below.

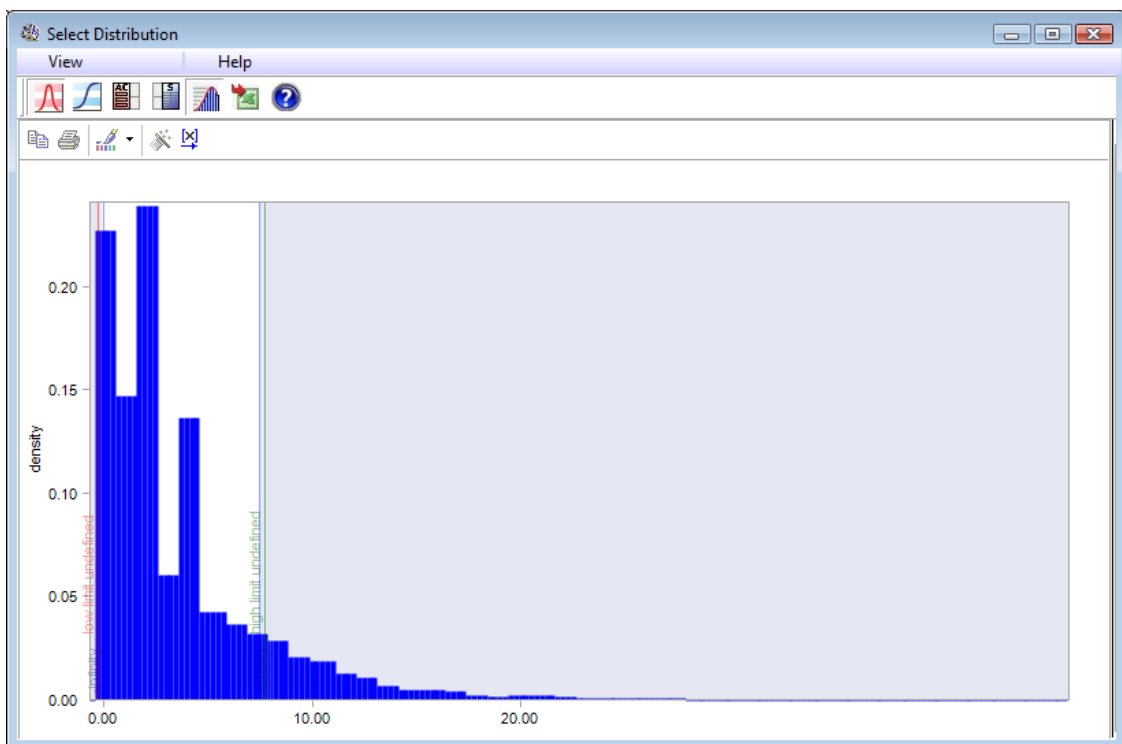


Figure 7 – Histogram of total risk

Users of other Monte Carlo simulation software packages may think that the above example model is not really that different from what is possible with other software. There are, however, a few important differences:

- The VoseRiskEvent(Probability, Impact,  $U$ ) function creates a distribution that combines the probability and impact together. Driving the sampling of the distribution via a Uniform(0,1) in the host Monte Carlo simulation software (in this case @RISK) allows you to perform a correct *sensitivity analysis* of your project. For example, we get the Tornado chart of Figure 2:

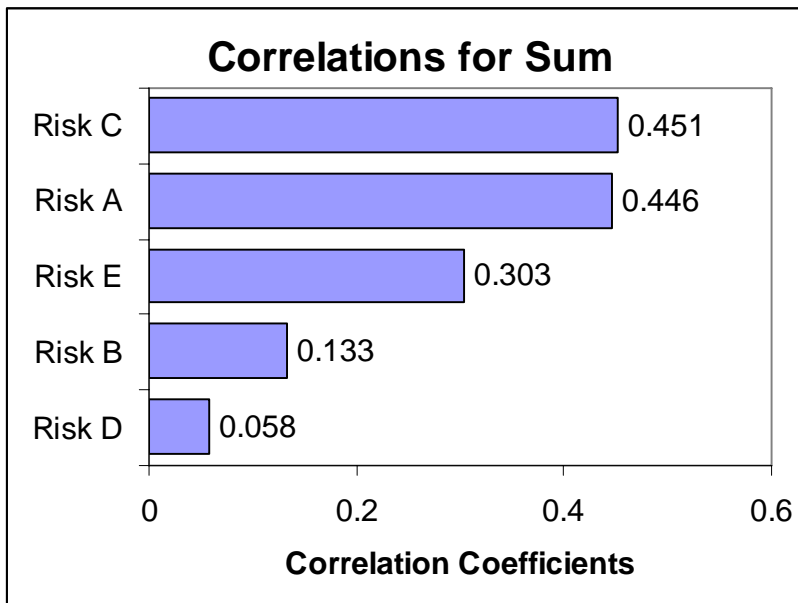


Figure 2- Tornado chart for the Risk Event model with ModelRisk

If we had done this model with just the tools available with @RISK (see Figure 3), or any other Monte Carlo simulation software package, we would get the Tornado chart of Figure 4.

	A	B	C	D	E	F
1						
2		Risk Event	Probability	Impact	Event	
3		A	0.2	3.27105412	3.2711	
4		B	0.1	7.14823065	7.1482	
5		C	0.4	2.515289597	0	
6		D	0.02	9.370751868	0	
7		E	0.5	1.753673828	1.7537	
8				Sum	12.1730	
9						
10		<b>Formulae table</b>				
11		D3 (D4:D7 similar)	=RiskLognorm(6,3)			
12		E3:E7	=RiskBinomial(1,C3)*D3			
13		E8	=RiskOutput() + SUM(E3:E7)			
14						

Figure 3 - Risk Event model with @RISK

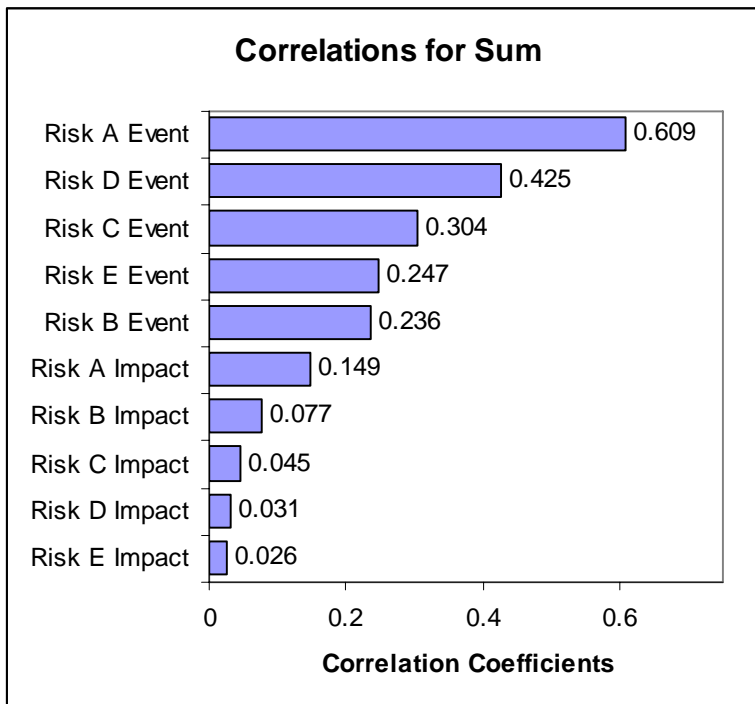



Figure 4 - Tornado chart for the Risk Event model with @RISK

- Using ModelRisk you can now see the impact of the whole risk (probability and impact combined for example Risk A), and not the separate effect of the probability of Risk A and the impact of Risk A;
- For anyone constructing or reviewing the risk model, it is very easy to see what is being modeled since the model is compact and easy to follow.
- Finally, while the above model is fairly simple, it would be relatively easy to expand and enhance this model, without making the modeling very complex.

#### 4.1.6 Conclusion

Using these results, the project manager can give all stakeholders a realistic view of the probability of going over budget due to these risks as well as the possible impact of going over budget. In addition, using the sensitivity analysis you can determine the main risk drivers (i.e. ‘what factors are causing my project to go over budget?’), and therefore make better decisions on how to reduce the risks of the project.

While the above example was fairly simple, it demonstrates several very useful features of *ModelRisk*:

- Distributions and Objects
- Risk Events (combination distribution of probability and impact);
- Monte Carlo simulation
- The View Function button 

The next example will show you how to use some of the time series fitting and modeling capabilities within ModelRisk.

## 4.2 Time Series Fitting and Modeling

### 4.2.1 Problem Description

Imagine that you work for a utility company. Your company purchases large quantities of a commodity (e.g. coal or natural gas) in order to operate their facilities. One of your new responsibilities is to forecast the future price of this commodity. You currently have historical monthly prices from the last 30 months which are shown below in Table 2. You would like to build a model that forecasts commodity prices for six months into the future.

Month/Year	2005	2006	2007
January	117	138	160
February	127	147	154
March	129	161	148
April	147	163	165
May	139	152	161
June	134	147	156
July	126	149	?
August	124	156	?
September	123	136	?
October	129	135	?
November	135	134	?
December	147	136	?

Table 2 – Historical commodity prices

### 4.2.2 Conceptual Solution

You know from previous experience that the prices for this commodity tend to follow a time-series pattern in the family of Geometric Brownian Motion with Mean Reversion (GBMMR). You therefore decide to use this knowledge and build a time-series model to forecast the price for the next six months.

First, you need to fit a GBMMR time-series model to your historical data. After this, we can use that model to forecast prices for six months into the future. We explain these two steps in the next two sections.


### 4.2.3 Fitting the GBMMR time-series Model to your data

Here are the steps to build this model using *ModelRisk*:

- First enter the data shown in the chart above into Excel with the top left in cell A1. The result should look similar Figure 8 below.

	A	B
1	<b>Mth-Yr</b>	<b>Price</b>
2	Jan-05	117
3	Feb-05	127
4	Mar-05	129
5	Apr-05	147
6	May-05	139
7	Jun-05	134
8	Jul-05	126
9	Aug-05	124
10	Sep-05	123
11	Oct-05	129
12	Nov-05	135
13	Dec-05	147
14	Jan-06	138
15	Feb-06	147
16	Mar-06	161
17	Apr-06	163
18	May-06	152
19	Jun-06	147
20	Jul-06	149
21	Aug-06	156
22	Sep-06	136
23	Oct-06	135
24	Nov-06	134
25	Dec-06	136
26	Jan-07	160
27	Feb-07	154
28	Mar-07	148
29	Apr-07	165
30	May-07	161
31	Jun-07	156
32	Jul-07	
33	Aug-07	
34	Sep-07	
35	Oct-07	
36	Nov-07	
37	Dec-07	

Figure 8 – Excel table of historical prices

- Now select the data, click on the Time Series Fit button  and you should see the following window shown in Figure 9 appear:

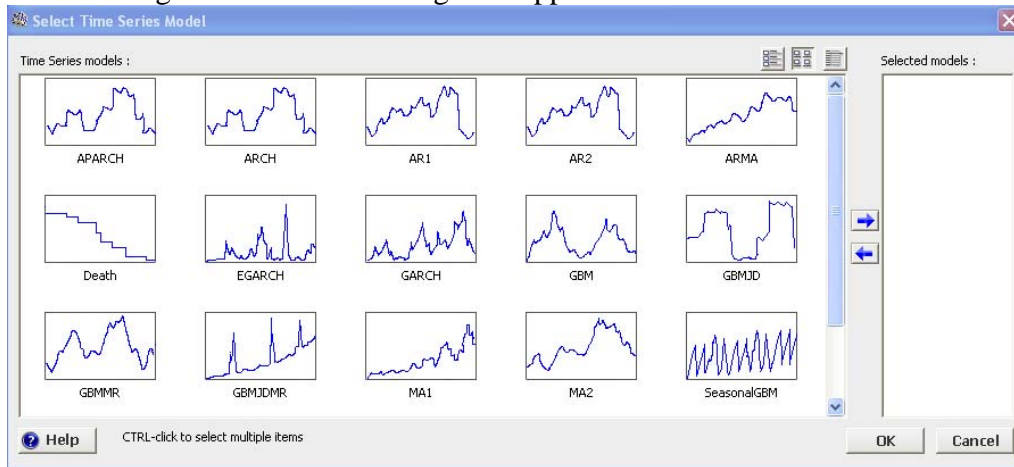


Figure 9 – Select Time Series window

- In the “Select Time Series Model” window, double-click GBMMR and click OK.
- In the “PastTimeStamps” field enter the historical timestamp range, A2:A31, or click the button next to the field and highlight the data range in the spreadsheet.
- In the “FutureTimeStamps” field enter the forecast timestamp range, A32:A37, or click the button next to the field and highlight the data range in the spreadsheet.
- In the “Output Location” field enter the range of cells for our forecast, B32:B37, or click the button next to the field and highlight the range of cells in the spreadsheet.
- The Time Series Fit window should now look similar to Figure 10:

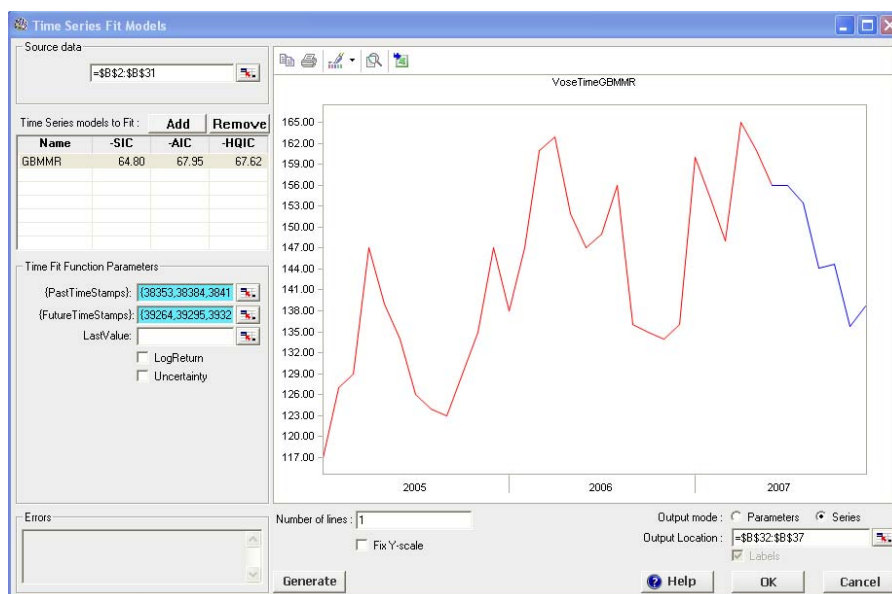


Figure 10 - GBMMR data fitted with forecast

- The red portion of the graph is the historical data and the blue portion is one instance of the forecast for the next six months. If you hit the enter key or click the **Generate** button, you will see that the red portion is stationary but that the blue portion will change. Each time you do this you are generating a new scenario (‘iteration’) of potential future prices.
- It is difficult to see how a stochastic forecast behaves by viewing just one forecast scenario at a time. In the “Number of lines” field enter the value 10. The window should look similar to Figure 11 showing a set of 10 forecasts.

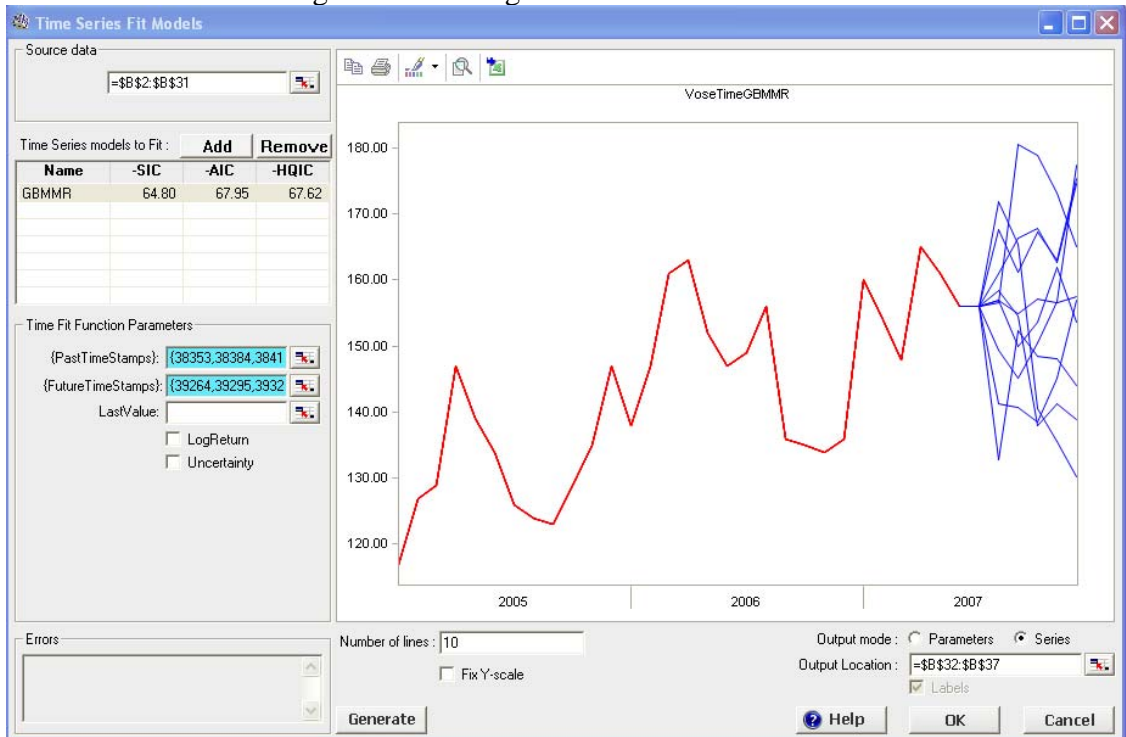




Figure 11 – GBMMR data fitted with multiple forecast lines

- When you click OK, you can see that the cells for the future six months are now populated with the forecasted values (Figure 12):

	A	B
1	<b>Mth-Yr</b>	<b>Price</b>
2	Jan-05	117
3	Feb-05	127
4	Mar-05	129
5	Apr-05	147
6	May-05	139
7	Jun-05	134
8	Jul-05	126
9	Aug-05	124
10	Sep-05	123
11	Oct-05	129
12	Nov-05	135
13	Dec-05	147
14	Jan-06	138
15	Feb-06	147
16	Mar-06	161
17	Apr-06	163
18	May-06	152
19	Jun-06	147
20	Jul-06	149
21	Aug-06	156
22	Sep-06	136
23	Oct-06	135
24	Nov-06	134
25	Dec-06	136
26	Jan-07	160
27	Feb-07	154
28	Mar-07	148
29	Apr-07	165
30	May-07	161
31	Jun-07	156
32	Jul-07	159
33	Aug-07	170
34	Sep-07	179
35	Oct-07	173
36	Nov-07	167
37	Dec-07	168

Figure 12 – Spreadsheet with forecasted values in red.

- As in the previous example model, each time you hit the F9 key, *ModelRisk* will generate a new instance of the time-series forecast for each of the next 6 months. If you wish to view or make changes to the fit, select any of the six forecast cells and click the  button or the  button, make any changes and click OK.

#### **4.2.4 Running the Model**

- In this example, imagine that you are most interested in the probability that the commodity price in Dec-07 will be greater than 190.
- In your Monte Carlo simulation software, designate the forecast cell of Dec-07 (B37) as an output/forecast and run the model for at least 10,000 iterations. From the resulting probability distribution of prices on Dec-07, you can now determine the probability that the price will be greater than 190.
- You will see that based on this model, there is roughly a 17.2% probability that the price in Dec-07 will be greater than 190.

#### **4.2.5 Conclusion**

From this example we have learned some of the basic features of *ModelRisk's* capabilities in Time Series Fitting and forecasting and how easy it is to use *ModelRisk* to include time series features in your models. As we mentioned before, you can find a lot more information about Time Series Fitting and modeling (including some truly advanced time series methods) in the ModelRisk Help File.

### 4.3 Simple Aggregate Distribution Model

In this very simple example, we show how to generate an aggregate distribution from two other distributions. There are several aggregate distribution methods available with *ModelRisk*. In this example we will demonstrate two methods.

#### 4.3.1 Problem Description

Imagine that you are an analyst at an insurance company. You would like to build a distribution of the potential losses for the next 12 months from a particular automobile accident policy.

We know that the insurance company has 464 policy holders and that the expected number of claims per policy holder per year is 0.245 (we have estimated this from historical data). We assume that the number of claims per year will follow a Poisson distribution with a rate of  $464 * 0.245 = 113.68$ . We also know that the size of historical claims follow a Lognormal distribution with a mean of US\$2,400 and a standard deviation of US\$900. This particular policy has a deductible of US\$2,000 which means that the insurance company will only have to pay the portion of a claim where the damage exceeds US\$2,000. Policyholders will therefore only file a claim if the total damage is greater than US\$2,000.

#### 4.3.2 Conceptual Solution

Conceptually, what we have to do here is find the total of a random number of random variables. The number of random variables to be added together is called the '*frequency distribution*'. In our case that is the number of claims in the next 12 months, which we will model as a Poisson distribution with a rate of 113.68. The size of each random variable to be summed is called the '*severity distribution*'. In this example, the severity distribution is the Lognormal with a mean of US\$2,400 and a standard deviation of US\$900, where the distribution is also truncated below US\$2,000.

Frequent mistake: In our experience of auditing models, summing random variables is one of the areas where even experienced risk analysts often make mistakes in their modeling. A frequent mistake in risk modeling is to take a single sample from the frequency distribution and multiply it by a single sample from the severity distribution and simulate it for a large number of iterations. For example, if in one iteration the random sample from the claim frequency is 100 and the sample from the severity distribution is US\$2,500, we might assume that the total claim losses for this iteration for next year to be  $100 * (US\$2,500 - 2,000) = US\$50,000$ . This seems intuitively correct at first glance but it is actually an incorrect technique. Using this method in a model assumes that for this iteration each of the claims for next year has damage of exactly US\$2,500 which is clearly not realistic. In reality, each of the 100 claims will likely have a different severity.

Correct method: The correct way to model this scenario is to first take a sample from the frequency distribution which is the number of claims and then take a sample from the severity distribution for each claim. For example, if the sample from the frequency distribution is 100, we would need to take 100 samples from the severity distribution and sum them to get an estimate of claim losses for the next year. Then this would need to be done for several thousand iterations to build the aggregate distribution of total claims for the next 12 months.

With the parameters used in this example, this correct method is a feasible option. However, when the frequencies are large (i.e. thousands of accidents/month), it is very cumbersome and time consuming to implement this procedure for building an aggregate distribution in a standard Monte Carlo simulation package. For each aggregate distribution, a number of steps using conditional statements and look up tables is required. In a real-world model with many aggregate distributions, this could result in a very large spreadsheet model (possibly thousands of rows) which will be prone to errors and difficult to debug, or even impossible to implement.

In contrast, *ModelRisk* incorporates several very powerful techniques available to provide simple and intuitive methods for modeling aggregate distributions. The methods available in *ModelRisk* include the aggregated Monte Carlo method, recursive methods and Fast Fourier Transform (FFT) methods. For more details on these, please refer to the *ModelRisk* Help file.

We will first choose the `VoseAggregateMC()` function which replicates the Monte Carlo simulation procedure described above (under the heading ‘correct method’) but in a much easier, faster and more straightforward way.

We will also show how to use FFT technology built into *ModelRisk* to implement this model with the `VoseAggregateFFT()` function.

### 4.3.3 Building the Model

#### 4.3.3.1 VoseAggregateMC() Model

- Open *ModelRisk* and Excel.
- Starting in cell A1 enter a table like the one shown here in Figure 13.

	A	B	C	D	E	F
1	Policy holders	Expected claims per year	Damage mean	Damage stdev	Deductible	Total Claim \$
2	464	0.245	2400	900	2000	

*Figure 13 – Aggregate model data*

- Select Cell F2 and then from the Vose *ModelRisk* drop-down menu select Aggregate modeling>>Aggregate MC.
- In the “Frequency Distribution” field, enter “VosePoisson(A2\*B2).
- In the “Severity Distribution” field, enter “VoseLognormalObject(C2,D2,,VoseXBounds(E2,))”.

- When the VoseXBounds(min,max) function is used in a distribution statement, it specifies where to truncate a distribution on the high and low end. If one of the fields is left blank, VoseXBounds defaults to +/- infinity respectively.
- Notice that the default value in the Samples field is 1000. You will also notice that after you enter the Severity Distribution, *ModelRisk* will pause for at least several seconds. The reason for this is that *ModelRisk* is actually taking a sample from the Frequency Distribution and adding up that many samples from the Severity Distribution. This is being done 1000 times to build the aggregate distribution that you see on your screen.
- There are some distributions for which VoseAggregateMC() function can be determined by calculation rather than simulation. In those cases where calculation is possible, the function will run much faster. For all distributions that include a VoseXBounds() truncation, a simulation must be done so the function will be somewhat slower. For more details refer to the ModelRisk Help file.
- To complete this model, make sure the Output Location field has “=\$F\$2”. The window should look like Figure 13 below.

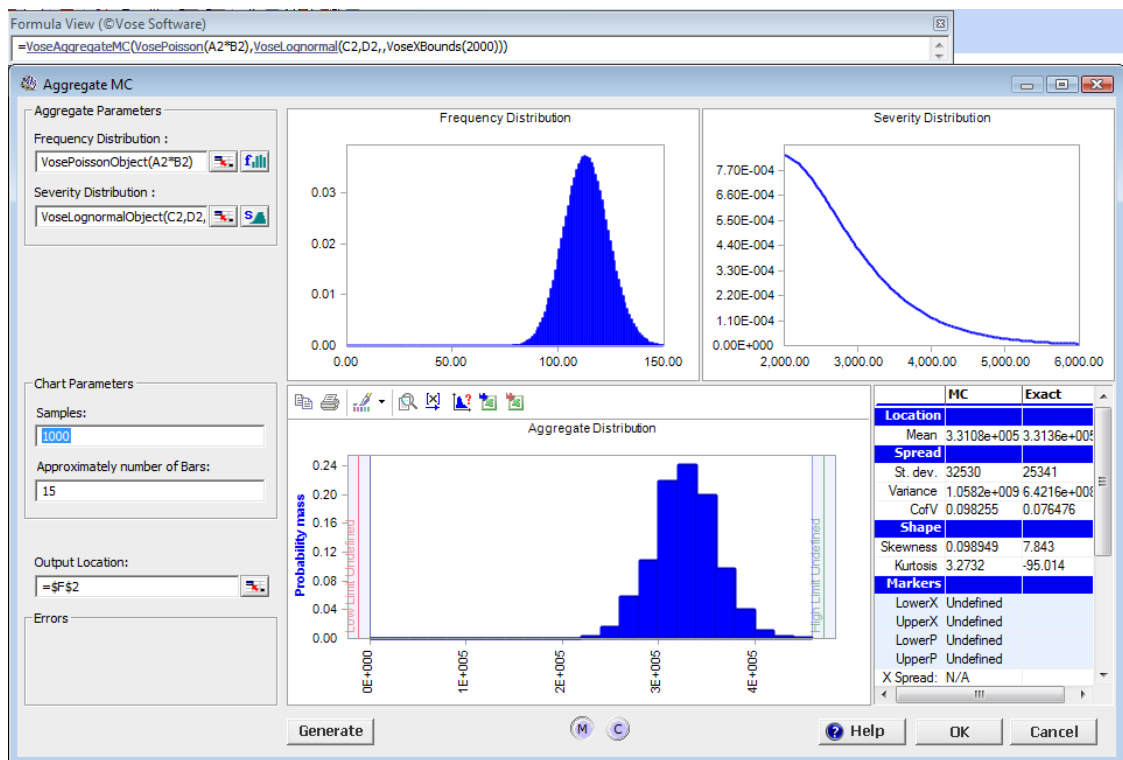


Figure 14 – VoseAggregateMC distribution window

- Now click the OK button.
- Your spreadsheet should look like Figure 15:

	A	B	C	D	E	F
1	Policy holders	Expected claims per year	Damage mean	Damage stdev	Deductible	Total Claim \$
2	464	0.245	2400	900	2000	309595

Figure 15 – Spreadsheet view from aggregate distribution

- Notice now that each time you hit the F9 key, *ModelRisk* displays a sample in cell F2 from the aggregate distribution immediately without the delay originally needed to build the distribution.


### 4.3.3.2 VoseAggregateFFT Model

Alternatively, you may choose to use a different technique for building aggregate distributions called the Fast Fourier Transform (FFT) method. The FFT method uses calculation techniques to estimate the aggregate distribution. The steps to use the Fast Fourier Transform (FFT) method in our example model are:

- In your original spreadsheet model, copy cells A2 through E2 and paste them into cells A3 through E3.
- Select cell F3 and enter  
`=VoseAggregateFFT(VosePoissonObject(A3*B3),VoseLognormalObject(C3,D3,,VoseXBounds(E3,)),)`
- Your spreadsheet should now look like Figure 16:

	A	B	C	D	E	F
1	Policy holders	Expected claims per year	Damage mean	Damage stdev	Deductible	Total Claim \$
2	464	0.245	2400	900	2000	310843.8081
3	464	0.245	2400	900	2000	350544.5023

Figure 16 – Spreadsheet view of MC and FFT aggregate distributions

- Select cell F3 and click the  button and the following window (Figure 17) will appear showing the Frequency Distribution, Severity Distribution and Aggregate Distribution (the aggregate distribution will be based on the FFT method);

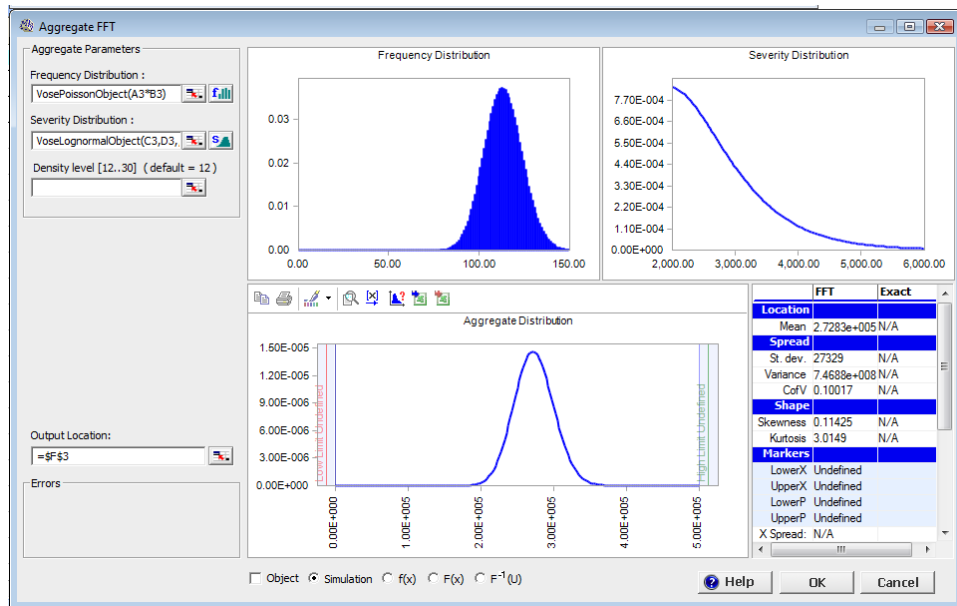


Figure 17 – VoseAggregateFFT distribution window

- Cells F2 and F3 will now sample from their respective distributions each time you push the F9 key.
- If you run a Monte Carlo simulation for several thousand iterations and track the results from both aggregate distributions, you find that the resulting distributions will be very similar.

#### 4.3.4 Conclusion

- In this third example we have demonstrated two fairly sophisticated risk modeling techniques of determining aggregate distribution that are (as you have seen) easy and faster to use with *ModelRisk*.
- You can now use samples from either of these aggregate distributions as part of a larger more complex model. One example might be trying to build an aggregate distribution for a number of different policies. You could do this by making a model similar to one of the above for each policy and the summing the result. If you were to run a Monte Carlo simulation on the total result, you could build an overall aggregate for the total losses from all of the policies. A spreadsheet model might look like Figure 18:

	A	B	C	D	E	F	G
1	<b>Policy</b>	<b>Policy holders</b>	<b>Expected claims per year</b>	<b>Damage mean</b>	<b>Damage stdev</b>	<b>Deductible</b>	<b>Total Claim \$</b>
2	A	464	0.245	2400	900	2000	302438
3	B	852	0.118	3560	1820	1500	395398
4	C	396	0.326	820	1260	250	133839
5	D	144	0.089	8250	11370	1000	132997
6	E	366	0.412	630	1280	100	133949
7						Total	1098621

Figure 18 – Example of determining the aggregate of several policies

- Finally, you may consider that the different insurance policies may be correlated which can be easily implemented with *ModelRisk* by using copulas.

## **5 Technical Support**

For more information about *ModelRisk* for Insurance and Finance,

- Web Site: <http://www.vosesoftware.com>
- Email: [info@vosesoftware.com](mailto:info@vosesoftware.com)
- Phone:
  - +32 (0)9 330 34 82 (Belgium)
  - +7 8672 753112 (Russia)
  - +1 303 440 8524 (United States) – Vose Consulting US