

OPNET/Riverbed Modeler: Viewing Simulation Results

Roman Dunaytsev

The Bonch-Bruевич Saint-Petersburg
State University of Telecommunications

roman.dunaytsev@spbgut.ru

Lecture № 7

1 Viewing results

2 Predefined filters

- As Is
- Probability Density (PDF)
- Probability Mass (PMF)
- Cumulative Distribution (CDF)
- Histogram (Sample Distribution)
- Histogram (Time Distribution)
- abscissa_filter
- adder
- average
- constant_shift
- delay_element
- differentiator
- exponentiator
- gain
- glitch_notch
- integrator
- limiter
- logarithm
- moving_average
- multiplier
- reciprocal
- sample_sum
- time_average
- time_window
- time_window_average
- value_notch

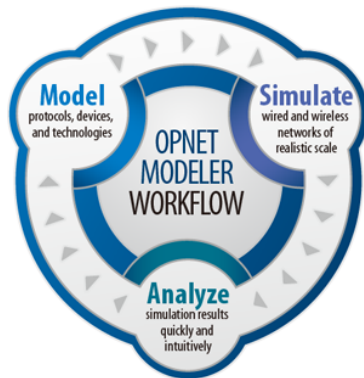
1 Viewing results

2 Predefined filters

- As Is
- Probability Density (PDF)
- Probability Mass (PMF)
- Cumulative Distribution (CDF)
- Histogram (Sample Distribution)
- Histogram (Time Distribution)
- abscissa_filter
- adder
- average
- constant_shift
- delay_element
- differentiator
- exponentiator
- gain
- glitch_notch
- integrator
- limiter
- logarithm
- moving_average
- multiplier
- reciprocal
- sample_sum
- time_average
- time_window
- time_window_average
- value_notch

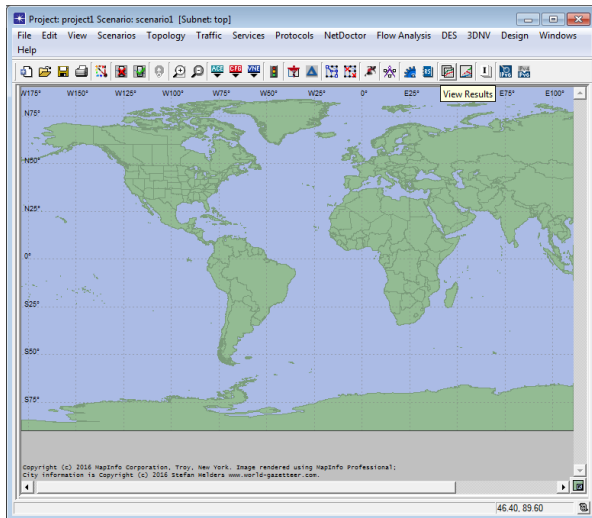
Viewing Results

- Once the simulation has finished running, it is time to display and analyze the collected results
- OPNET/Riverbed provides several utilities for displaying, examining, and comparing the results collected during simulation



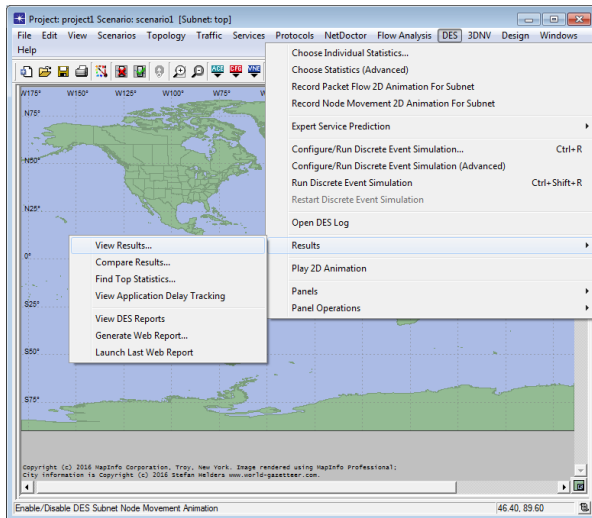
Viewing Results (cont'd)

- View Results



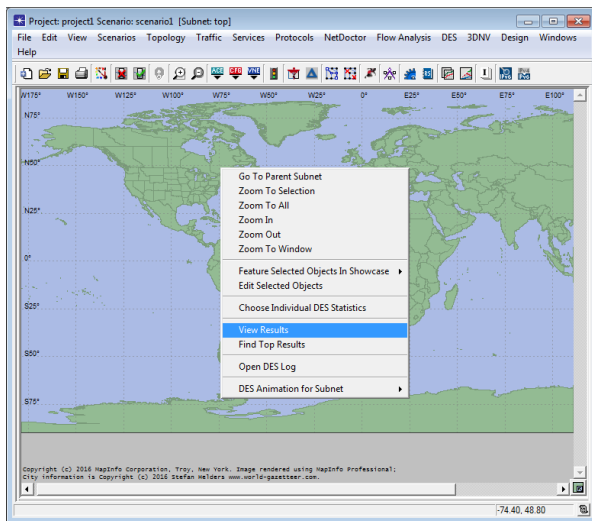
Viewing Results (cont'd)

- DES ⇒ Results ⇒ View Results...



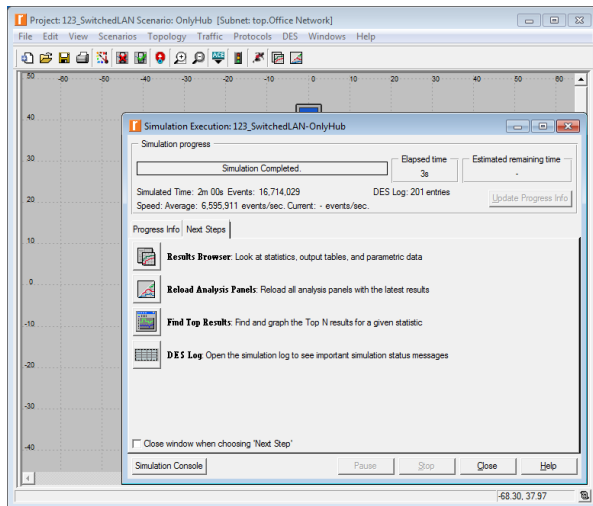
Viewing Results (cont'd)

- Right-click anywhere within the workspace \Rightarrow View Results



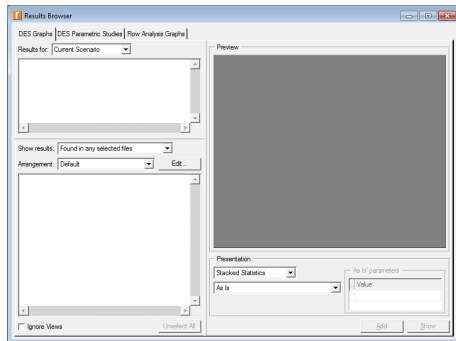
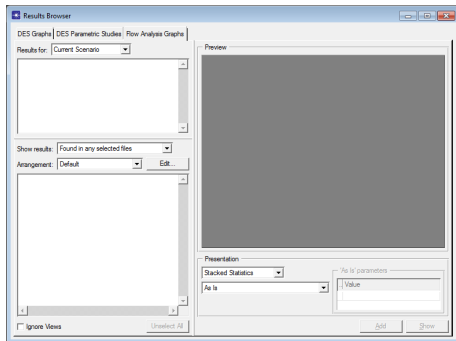
Viewing Results (cont'd)

- **Riverbed Modeler Academic Edition** : Results Browser



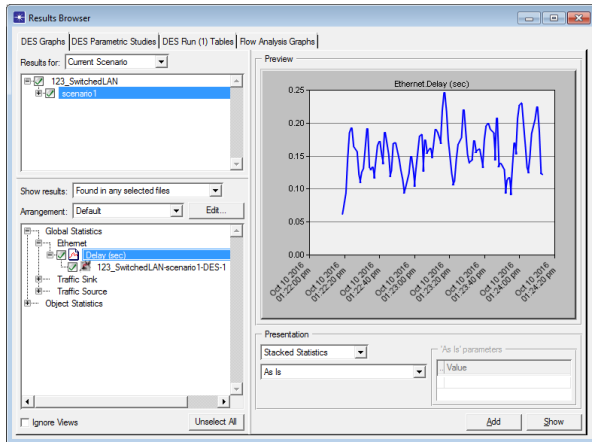
Viewing Results (cont'd)

- **Results Browser** – displays information in the form of graphs
- **OPNET Modeler vs. Riverbed Modeler Academic Edition**



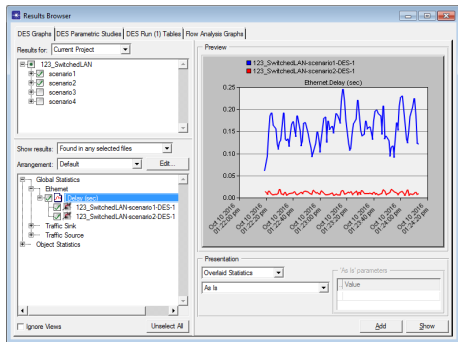
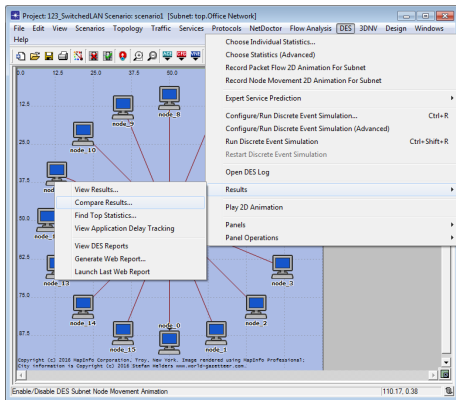
Viewing Results (cont'd)

- DES \Rightarrow Results \Rightarrow View Results... \Rightarrow **Current Scenario** (default)



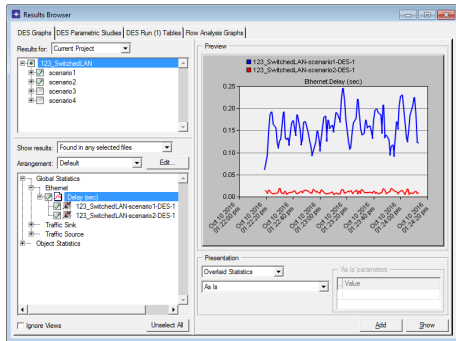
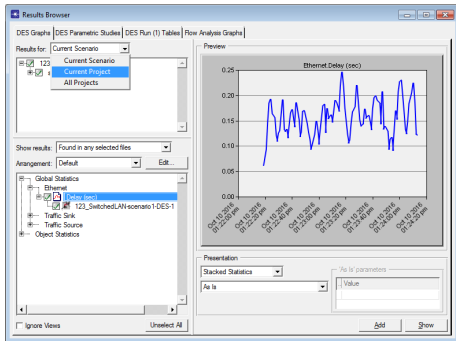
Viewing Results (cont'd)

- DES \Rightarrow Results \Rightarrow Compare Results...



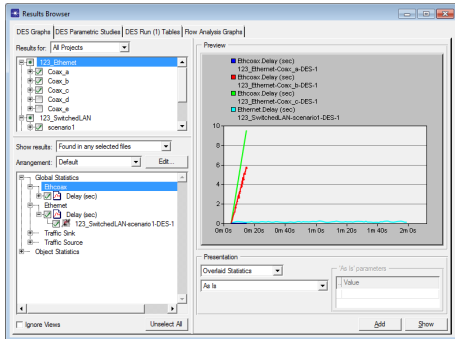
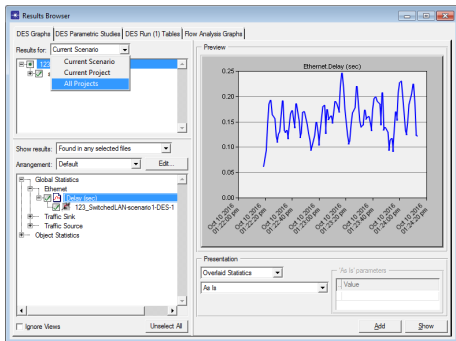
Viewing Results (cont'd)

- DES ⇒ Results ⇒ View Results... ⇒ **Current Project**



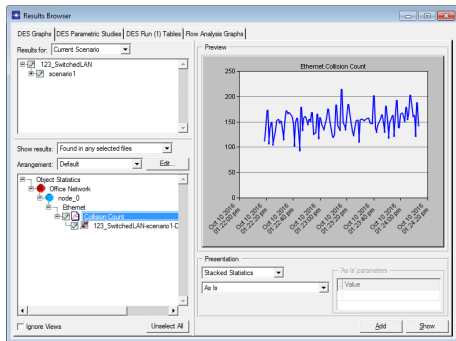
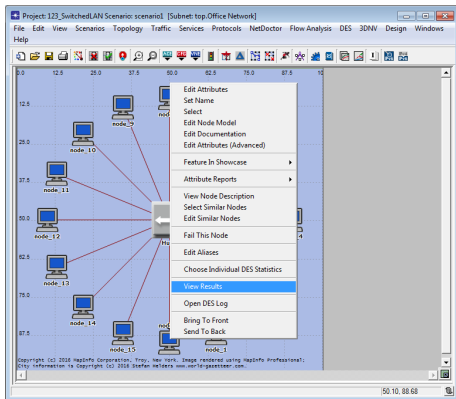
Viewing Results (cont'd)

- DES ⇒ Results ⇒ View Results... ⇒ **All Projects**
- **Results for** – specifies whether the simulation results will be retrieved from Current Scenario, Current Project, or All Projects (op_models)



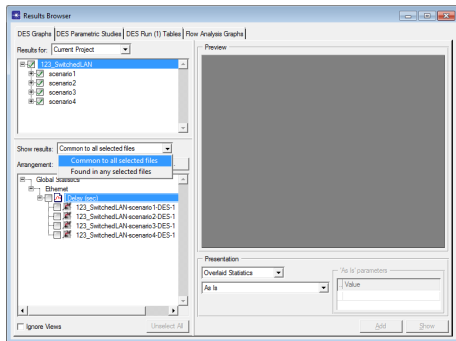
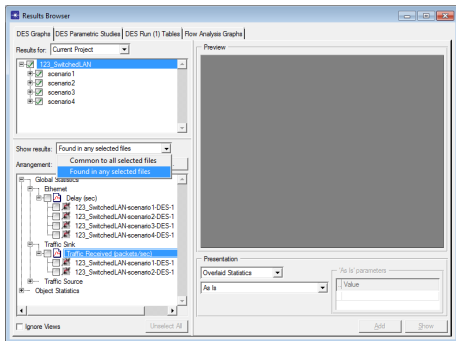
Viewing Results (cont'd)

- Viewing results for a single object
- Right-click on the object of interest \Rightarrow View Results



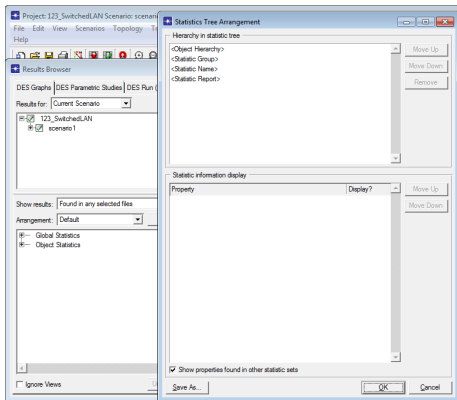
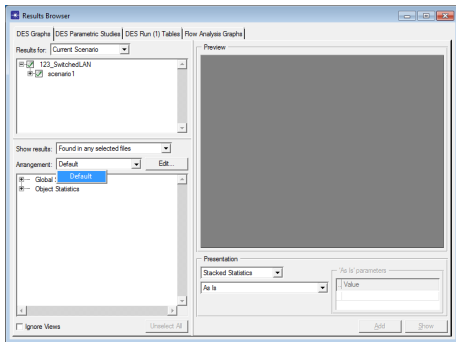
Viewing Results (cont'd)

- **Show results** – specifies which statistics will be available when multiple scenarios have been selected
- **Found in any selected file vs. Common to all selected files**



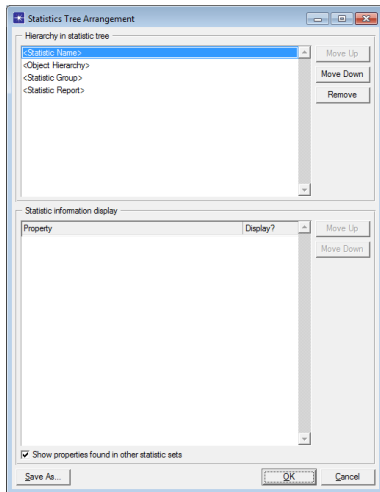
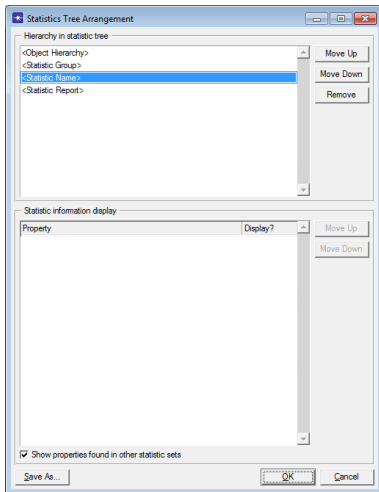
Viewing Results (cont'd)

- **Arrangement** – allows custom configuration of statistics arrangement



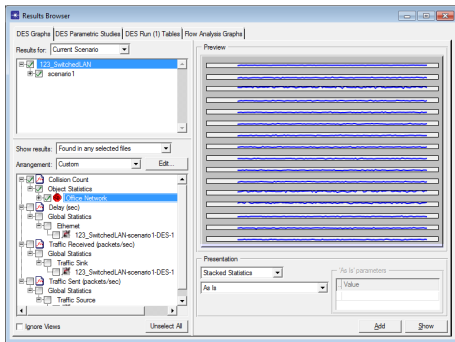
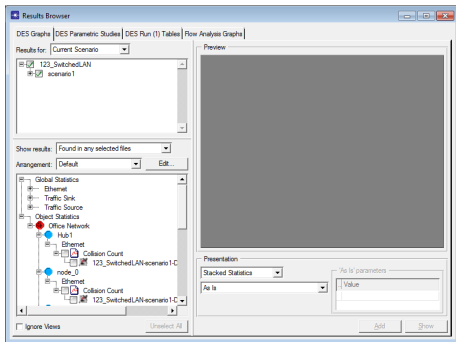
Viewing Results (cont'd)

- Arrangement: **Default** vs. **Custom**



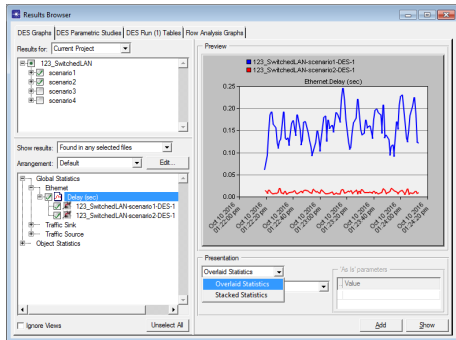
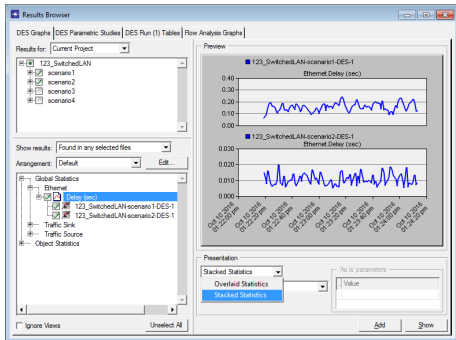
Viewing Results (cont'd)

- Arrangement: **Default** vs. **Custom**



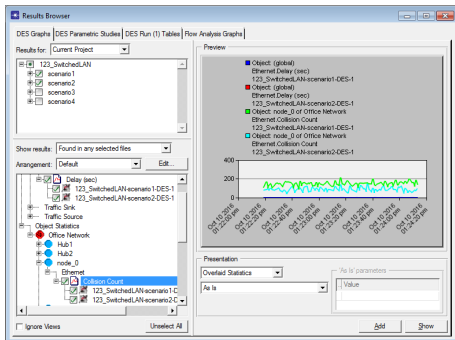
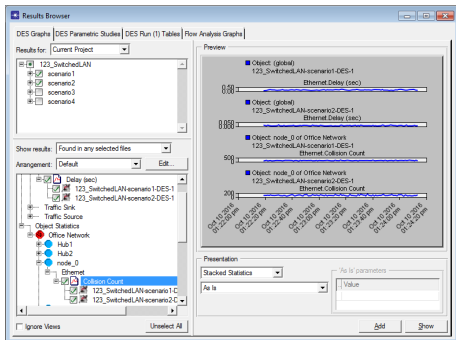
Viewing Results (cont'd)

- **Presentation** – controls the display of graphs when multiple statistics have been selected
- Presentation: **Stacked Statistics** vs. **Overlaid Statistics**



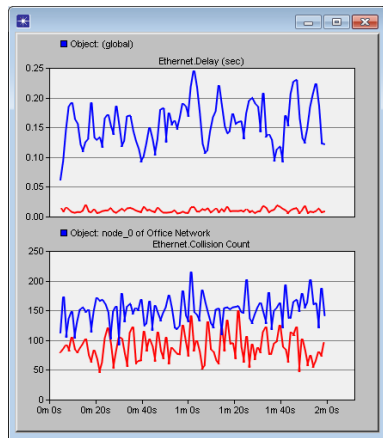
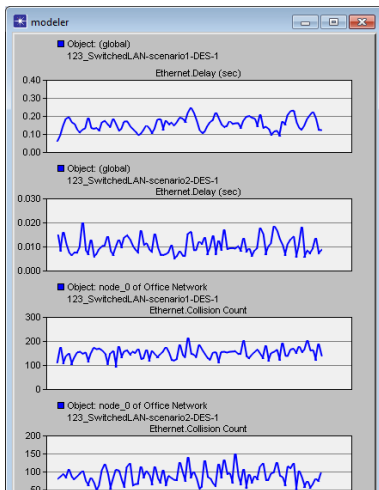
Viewing Results (cont'd)

- Don't mix different statistics on the same graph! ☹
- E.g., delay + load + retransmissions + ...



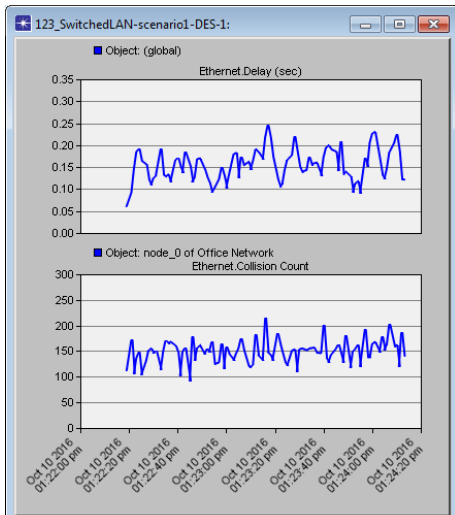
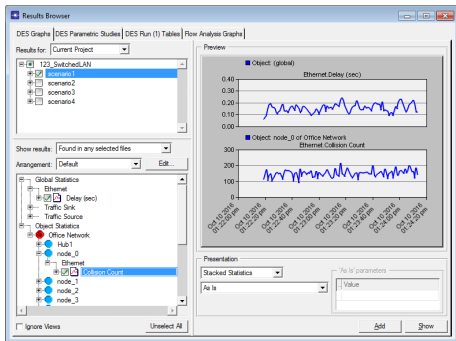
Viewing Results (cont'd)

- This is how it should be done
 - E.g., 2 scenarios, Delay (seconds) + Collision Count (node_0)



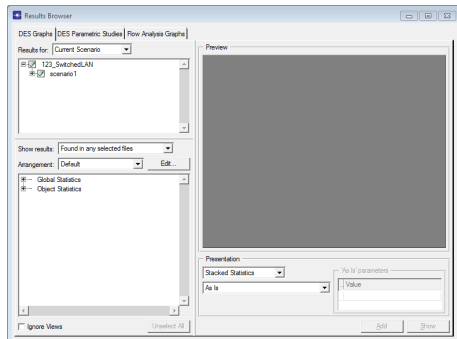
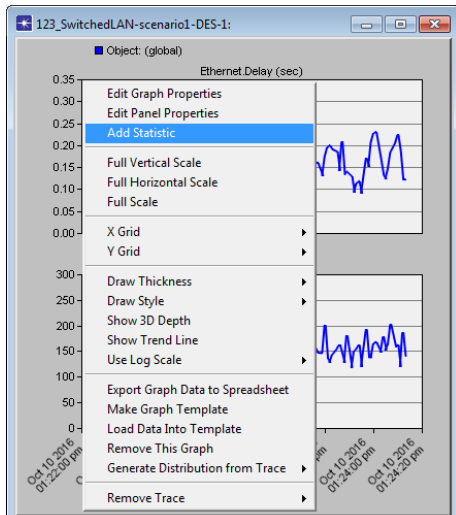
Viewing Results (cont'd)

- Adding graphs: Delay (seconds) vs. Collision Count (node_0)



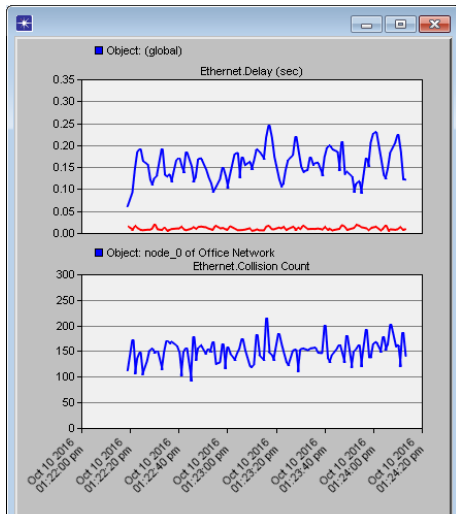
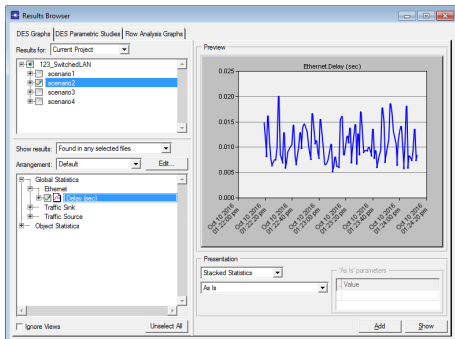
Viewing Results (cont'd)

- Adding graphs: Delay (seconds) vs. Collision Count (node_0)



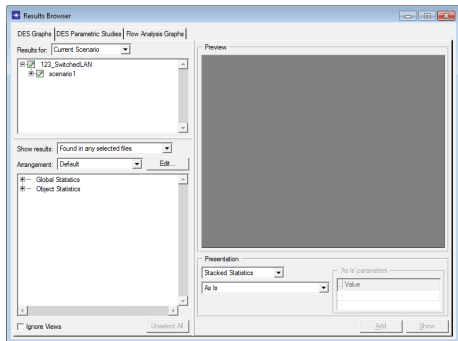
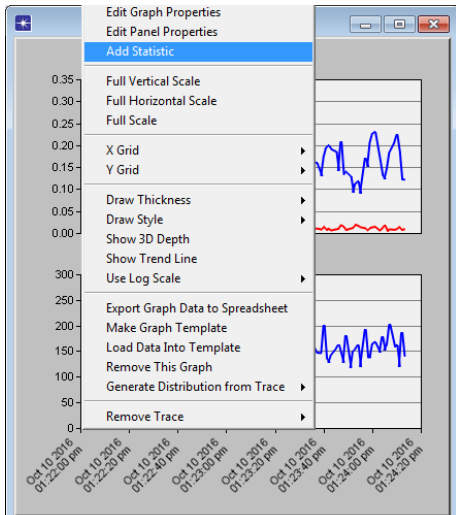
Viewing Results (cont'd)

- Adding graphs: Delay (seconds) vs. Collision Count (node_0)



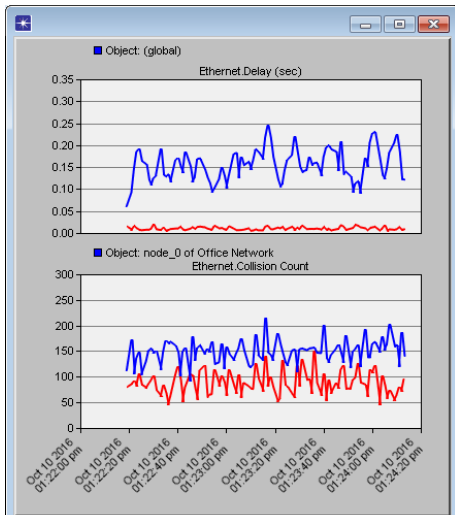
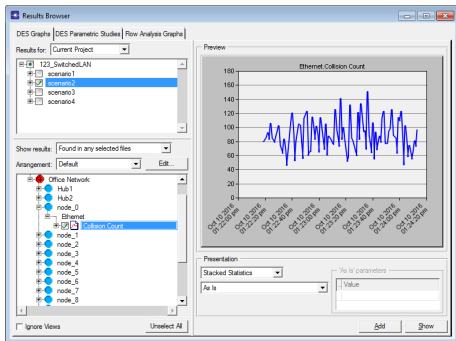
Viewing Results (cont'd)

- Adding graphs: Delay (seconds) vs. Collision Count (node_0)



Viewing Results (cont'd)

- Adding graphs: Delay (seconds) vs. Collision Count (node_0)



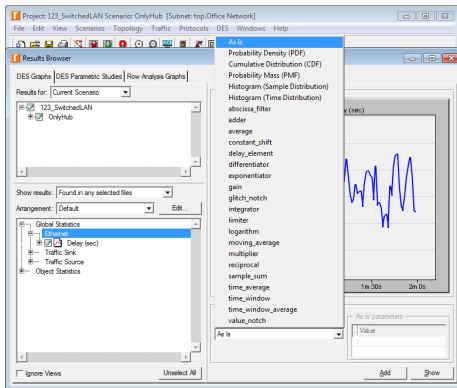
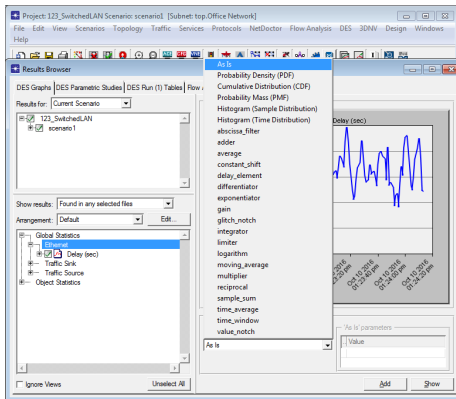
1 Viewing results

2 Predefined filters

- As Is
- Probability Density (PDF)
- Probability Mass (PMF)
- Cumulative Distribution (CDF)
- Histogram (Sample Distribution)
- Histogram (Time Distribution)
- abscissa_filter
- adder
- average
- constant_shift
- delay_element
- differentiator
- exponentiator
- gain
- glitch_notch
- integrator
- limiter
- logarithm
- moving_average
- multiplier
- reciprocal
- sample_sum
- time_average
- time_window
- time_window_average
- value_notch

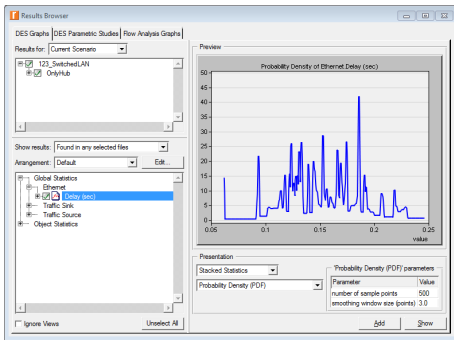
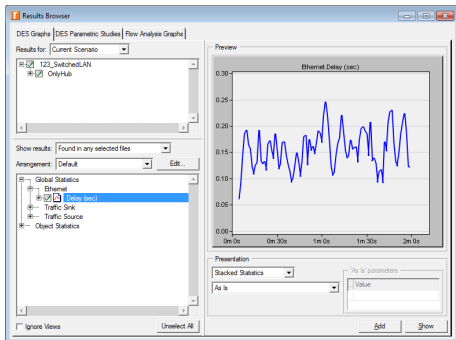
Predefined Filters

- **Predefined filters** – a set of operations that can be used to transform collected data to generate new statistics



Predefined Filters (cont'd)

- **Probability Density Function (PDF)** – corresponds to the likelihood that the input statistic's value lies within a specific range
 - $Pr(a \leq X \leq b) = \int_a^b f(x)dx$, where X is a **continuous(!)** RV



Predefined Filters (cont'd)

- $dx = \frac{\text{max value} - \text{min value}}{\text{number of sample points}} = \frac{0.246304161 - 0.061510018}{500} \approx 0.00037$ seconds
- Unlike a probability, a PDF can take on values greater than 1
- But its integral over the entire space is equal to 1

123_SwitchedLAN-OnlyHub-DES-1 - Excel

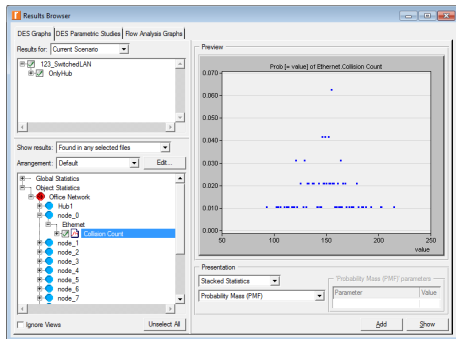
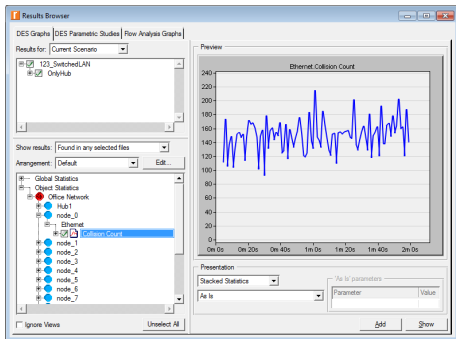
A	B
value	123_SwitchedLAN-OnlyHub-DES-1: Ethernet.Delay (sec)
0.06151	14.35482
0.06188	9.684381
0.062249	0.343501
0.062619	0.343501
0.062988	0.343501
0.063358	0.343501
0.063728	0.343501
0.064097	0.343501
0.064467	0.343501
0.064836	0.343501
0.065206	0.343501
0.065575	0.343501
0.065945	0.343501
0.066315	0.343501

123_SwitchedLAN-OnlyHub-DES-1 - Excel

A	B
0.24076	0.63194
0.24113	0.63194
0.2415	0.63194
0.241869	0.63194
0.242239	0.63194
0.242608	0.63194
0.242978	0.63194
0.243347	0.63194
0.243717	0.63194
0.244087	0.63194
0.244456	0.63194
0.244826	0.63194
0.245195	0.63194
0.245565	0.63194
2701.043	

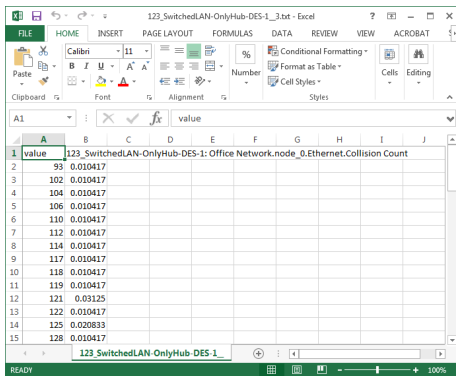
Predefined Filters (cont'd)

- **Probability Mass Function (PMF)** – corresponds to the likelihood that the input statistic's value is exactly equal to x
 - $Pr(X = x)$, where X is a **discrete(!)** RV



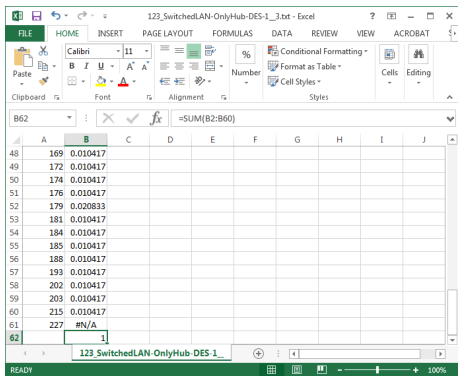
Predefined Filters (cont'd)

- $Pr(\text{Collision Count} = 155) = \frac{6}{\text{values per statistic} - \#N/A} = 0.0625$
 - Any event in the distribution has a probability between 0 and 1
 - The sum of all probabilities is 1



123_SwitchedLAN-OnlyHub-DES-1_3.txt - Excel

value	123_SwitchedLAN-OnlyHub-DES-1: Office Network_node_0.Ethernet.Collision Count
93	0.010417
102	0.010417
104	0.010417
106	0.010417
110	0.010417
112	0.010417
114	0.010417
117	0.010417
118	0.010417
119	0.010417
121	0.03125
122	0.010417
125	0.020833
128	0.010417

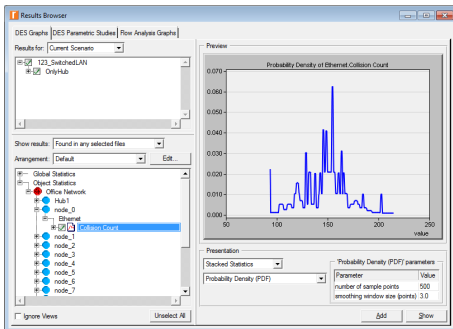
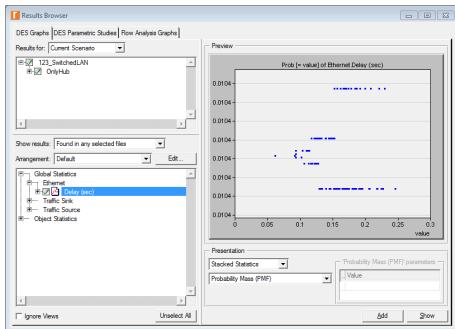


123_SwitchedLAN-OnlyHub-DES-1_3.txt - Excel

A	B	C	D	E	F	G	H	I	J
48	169	0.010417							
49	172	0.010417							
50	174	0.010417							
51	176	0.010417							
52	179	0.020833							
53	181	0.010417							
54	184	0.010417							
55	185	0.010417							
56	188	0.010417							
57	193	0.010417							
58	202	0.010417							
59	203	0.010417							
60	215	0.010417							
61	227	#N/A							
62		1							

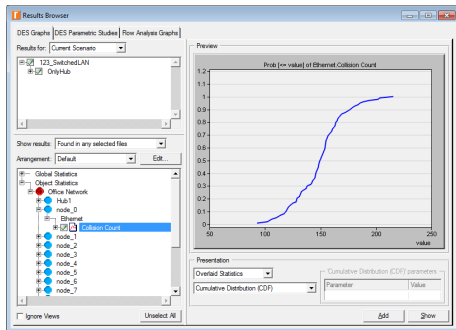
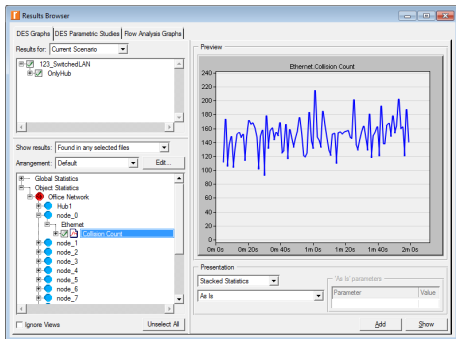
Predefined Filters (cont'd)

- Remember: PDF for continuous and PMF for discrete RVs!



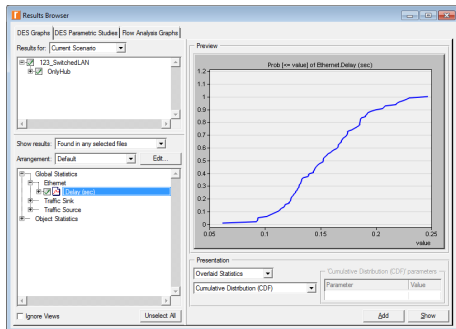
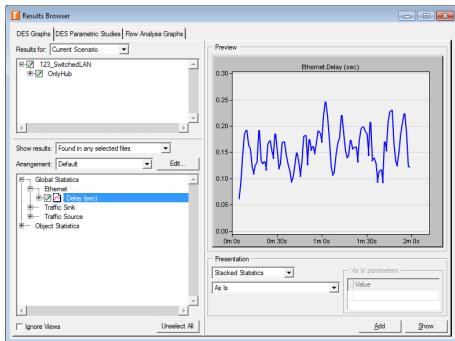
Predefined Filters (cont'd)

- **Cumulative Distribution Function (CDF)** – corresponds to the likelihood that the input statistic's value is less than or equal to x
 - $Pr(X \leq x)$, where X is a RV of **any(!)** kind



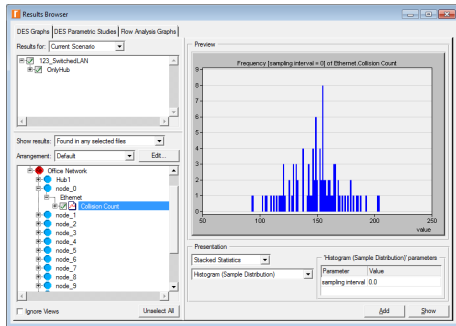
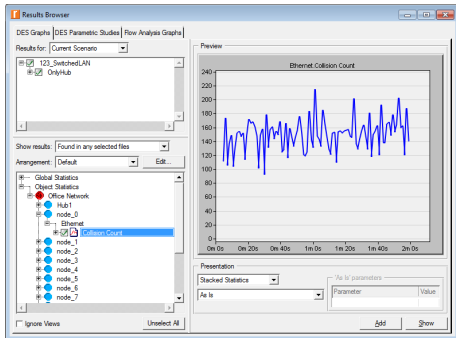
Predefined Filters (cont'd)

- In many cases, system performance requirements are stated as **'the probability of receiving a packet with delay larger than ... ms must be no greater than ... %'** and so on
 - CDF allows compliance with such a requirement to be readily determined by finding the threshold value on the horizontal axis and the corresponding probability on the vertical axis



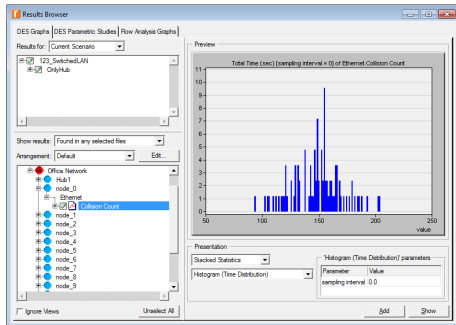
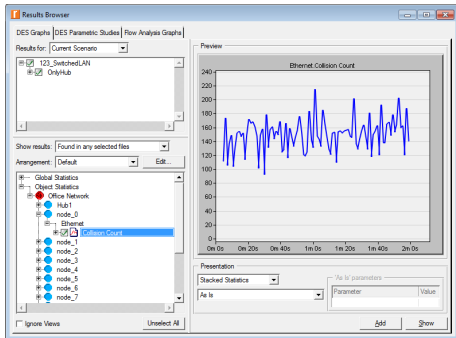
Predefined Filters (cont'd)

- **Histogram (Sample Distribution)** – uses the number of entries falling within each interval as the measure of frequency
 - If sample interval = 0.0, then there will be 100 intervals
 - If sample interval = . . . , then it sets the interval size



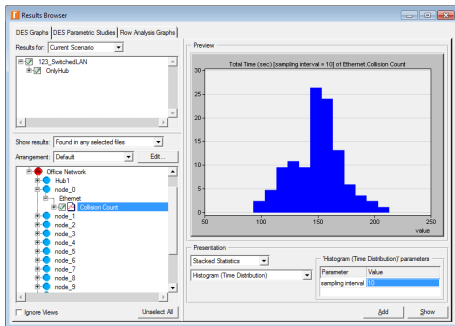
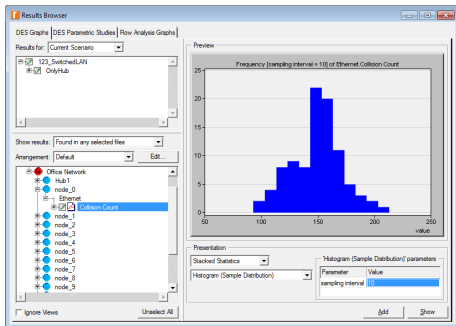
Predefined Filters (cont'd)

- **Histogram (Time Distribution)** – uses the 'time spent' by the statistic within each interval as the measure of frequency
 - If sample interval = 0.0, then there will be 100 intervals
 - If sample interval = . . . , then it sets the interval size



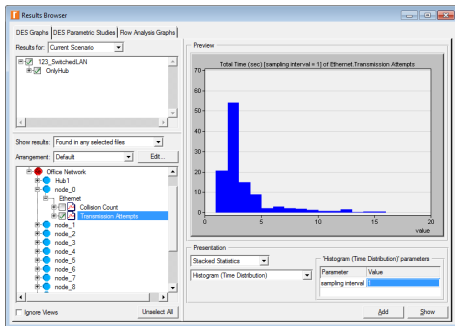
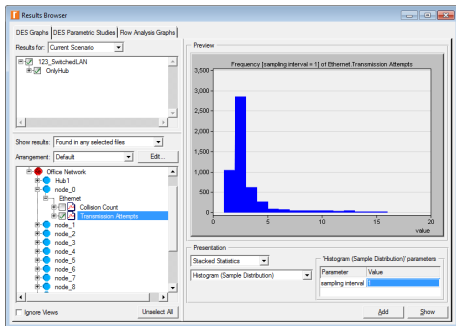
Predefined Filters (cont'd)

- Bucket mode: **Sample Distribution vs. Time Distribution**
 - For input statistics that have regularly spaced entries in terms of abscissa values, the shapes of the two histograms are identical



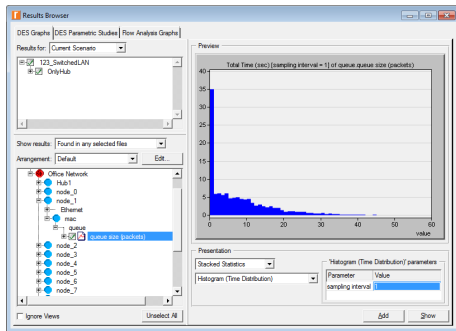
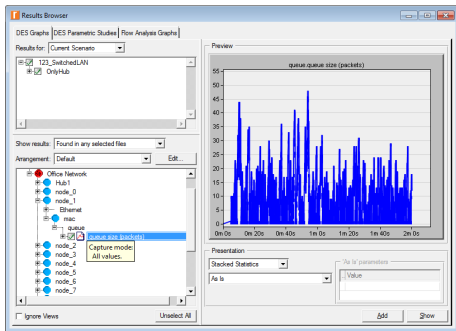
Predefined Filters (cont'd)

- All values: **Sample Distribution vs. Time Distribution**
 - For input statistics where abscissa values of entries are not regularly spaced, results can vary significantly between the two methods



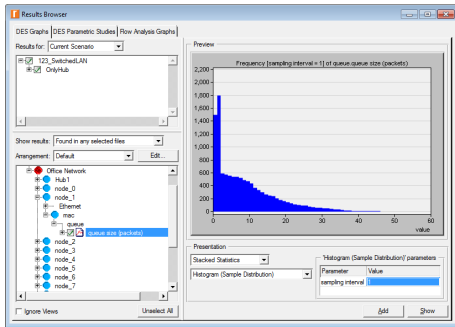
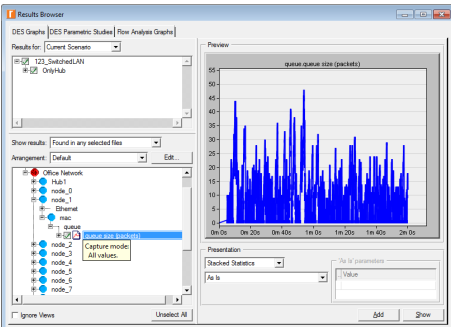
Predefined Filters (cont'd)

- **Histogram (Time Distribution)** – most appropriate for statistics that measure a quantity representing **the state information**
 - E.g., queue size, channel utilization, data rate
 - In case of queue size, it represents the amount of time that the queue size actually holds a particular value



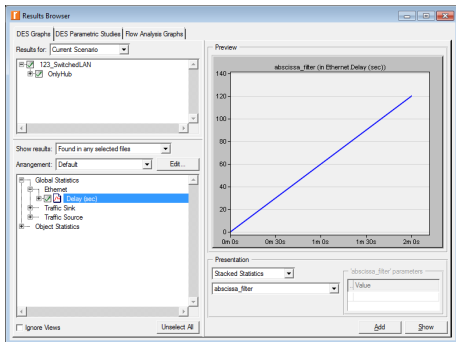
Predefined Filters (cont'd)

- **Histogram (Sample Distribution)** – most appropriate for statistics that measure a quantity representing **the occurrence of events**
 - E.g., packet delay, error rate, transmission attempts
 - In case of queue size, it represents how many times the queue size changed to then arrive at a particular value
 - This provides no definite information about how often one might expect to find the queue at a particular size ☹



Predefined Filters (cont'd)

- **abscissa_filter** – displays the abscissa values of a selected statistic
 - $\text{abscissa_filter}(i) = X_i$

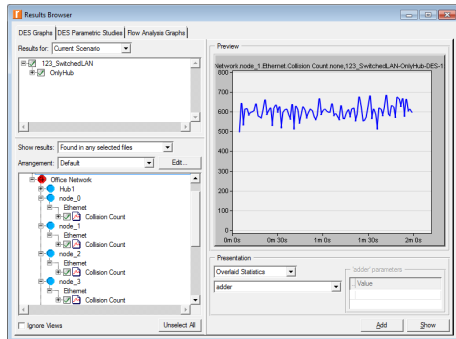
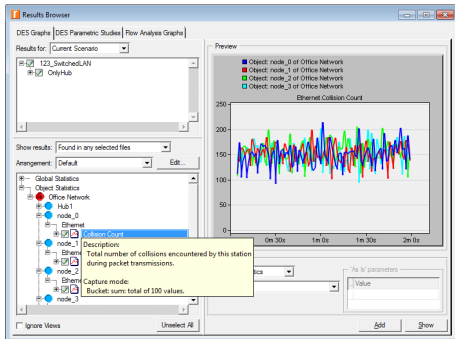


The screenshot shows an Excel spreadsheet titled '123_SwitchedLAN-OnlyHub-DES-1'. The data is organized in a table with columns A through J and rows 89 through 103. The table contains numerical values representing abscissa values over time. The values in column A are: 104.4, 105.6, 106.8, 108, 109.2, 110.4, 111.6, 112.8, 114, 115.2, 116.4, 117.6, 118.8, 120, 120. The values in column B are: 104.4, 105.6, 106.8, 108, 109.2, 110.4, 111.6, 112.8, 114, 115.2, 116.4, 117.6, 118.8, 120, 120. The value in cell A103 is #N/A. The spreadsheet is displayed in the 'READY' state with a zoom level of 100%.

	A	B	C	D	E	F	G	H	I	J
89	104.4	104.4								
90	105.6	105.6								
91	106.8	106.8								
92	108	108								
93	109.2	109.2								
94	110.4	110.4								
95	111.6	111.6								
96	112.8	112.8								
97	114	114								
98	115.2	115.2								
99	116.4	116.4								
100	117.6	117.6								
101	118.8	118.8								
102	120	120								
103	120	#N/A								

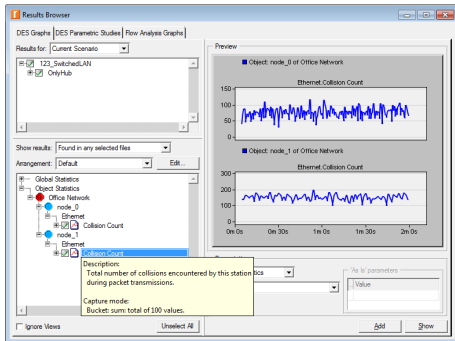
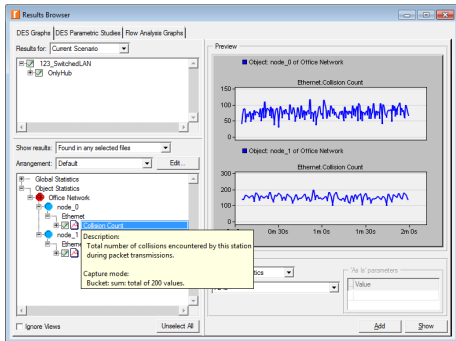
Predefined Filters (cont'd)

- **adder** – displays a graph that contains the sum of values of one or more statistics
 - If the input statistics have exactly the same number of entries and these entries are aligned with respect to their abscissa values, then the output statistic can be computed simply by adding ordinate values for entries of equal abscissa



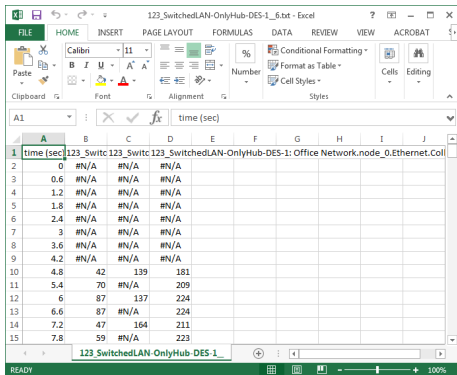
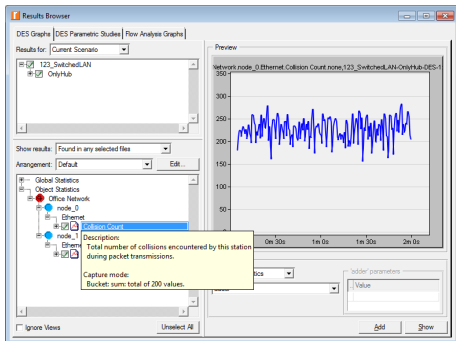
Predefined Filters (cont'd)

- What if the input statistics are not perfectly aligned with respect to each other?



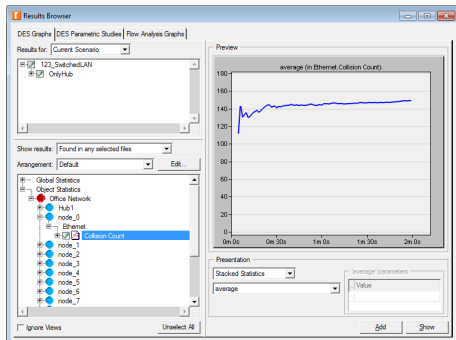
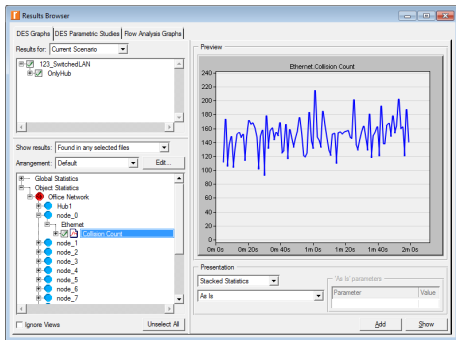
Predefined Filters (cont'd)

- In such a case, an abscissa alignment mechanism is automatically applied by this filter before adding is performed



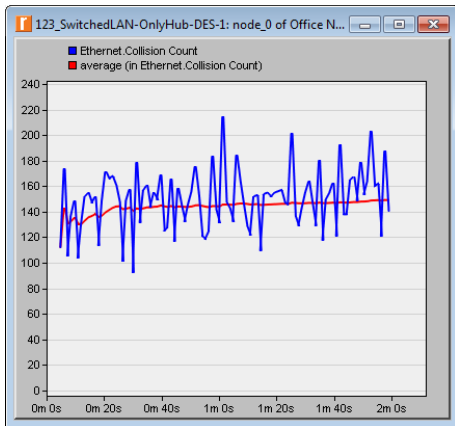
Predefined Filters (cont'd)

- **average** – displays the simple moving average of the values for a selected statistic



Predefined Filters (cont'd)

- $$\text{average}(n) = \frac{\sum_{i=0}^{n-1} Y_i}{n+1}$$

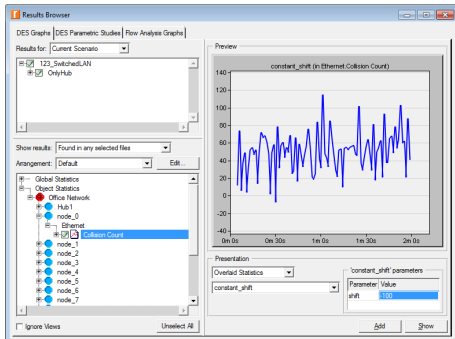
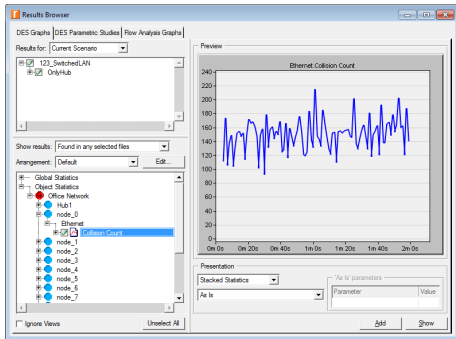


The screenshot shows an Excel spreadsheet with the following data:

time (sec)	123_Switch	123_SwitchEDS-1	123_SwitchEDS-1: Office Network.node_0.Ethernet.Collision Count
0	#N/A	#N/A	
1.2	#N/A	#N/A	
2.4	#N/A	#N/A	
3.6	#N/A	#N/A	
4.8	112	112	
6	174	143	
7.2	106	130.6667	
8.4	137	132.25	
9.6	149	135.6	
10.8	104	130.3333	
12	131	130.4286	
13.2	152	133.125	
14.4	155	135.5556	
15.6	147	136.7	

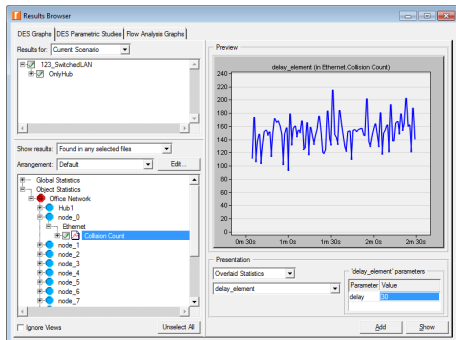
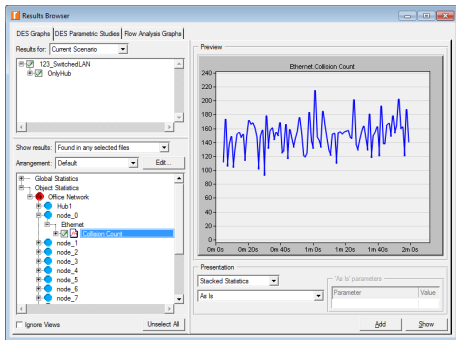
Predefined Filters (cont'd)

- **constant_shift** – shifts a selected statistic by a fixed amount ('shift') along the vertical axis



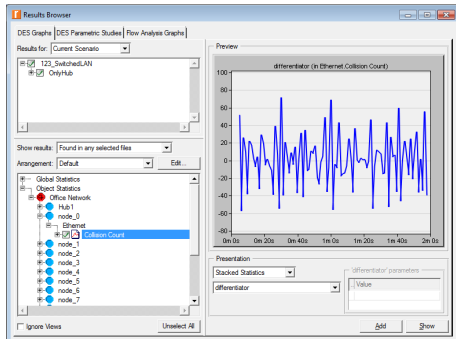
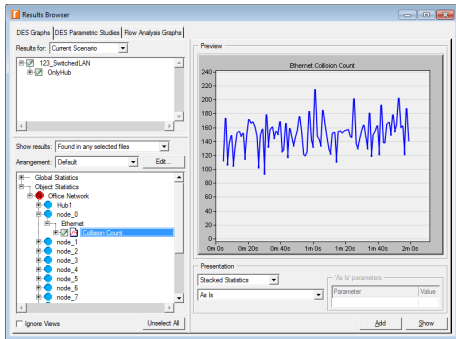
Predefined Filters (cont'd)

- **delay_element** – shifts a selected statistic by a fixed amount ('delay') along the horizontal axis



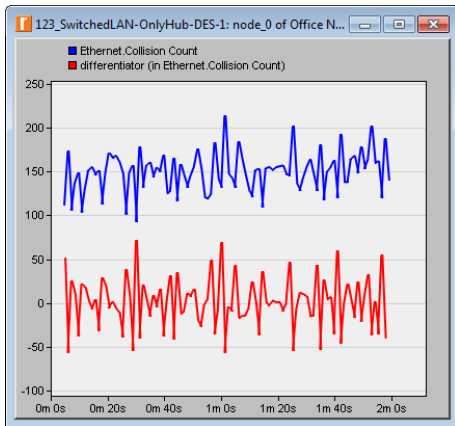
Predefined Filters (cont'd)

- **differentiator** – displays the derivative of a selected statistic with respect to its abscissa variable



Predefined Filters (cont'd)

- differentiator(i) = $\frac{Y_{i+1} - Y_i}{X_{i+1} - X_i}$

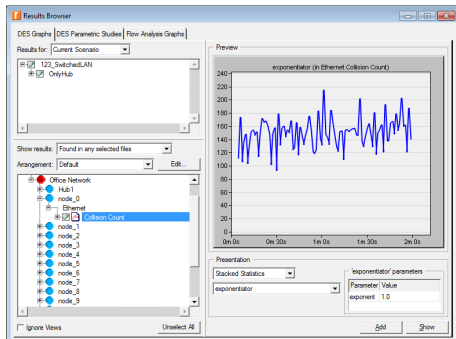
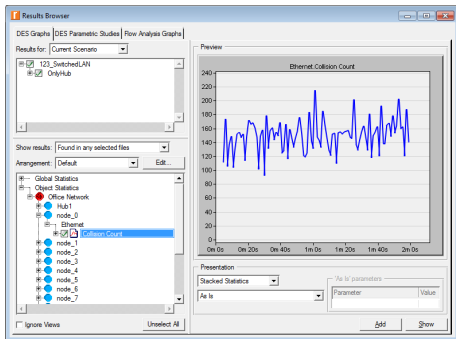


The Excel spreadsheet displays the data from the graph. The columns are labeled 'time [sec]', 'Ethernet.Collision Count', and 'differentiator (in Ethernet.Collision Count)'. The data points are as follows:

time [sec]	Ethernet.Collision Count	differentiator (in Ethernet.Collision Count)
0	#N/A	#N/A
1.2	#N/A	#N/A
2.4	#N/A	#N/A
3.6	#N/A	#N/A
4.8	112	51.66667
6	174	-56.66667
7.2	106	25.83333
8.4	137	10
9.6	149	-37.5
10.8	104	22.5
12	131	17.5
13.2	152	2.5
14.4	155	-6.66667
15.6	147	4.166667

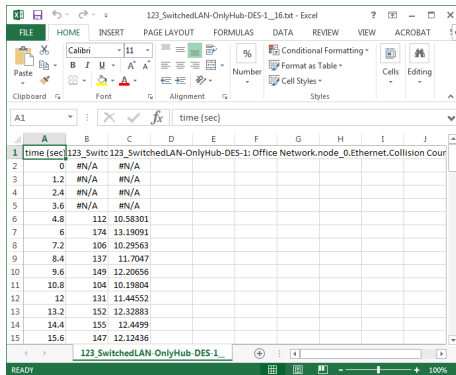
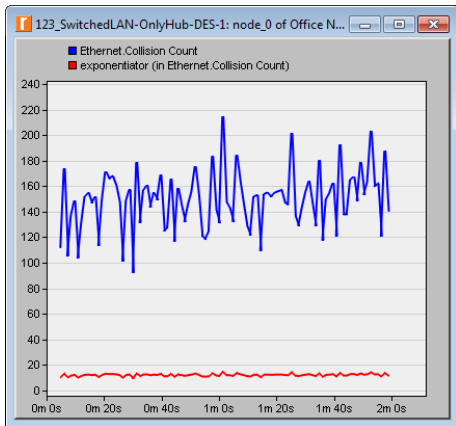
Predefined Filters (cont'd)

- **exponentiator** – raises the ordinate values of a selected statistic to a fixed power ('exponent')
 - If exponent = 1.0, then **exponentiator** = **As Is**



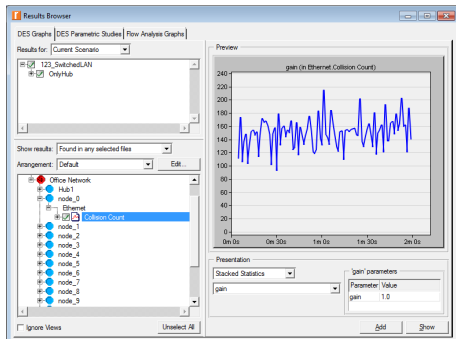
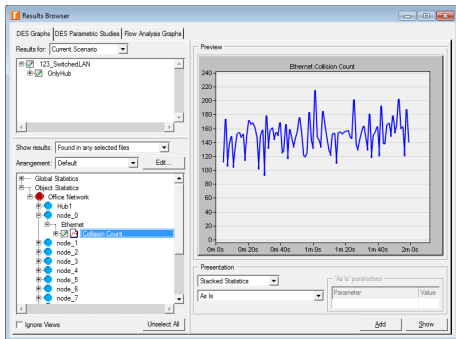
Predefined Filters (cont'd)

- exponentiator(i) = $(Y_i)^P$
 - E.g., $P = 0.5$



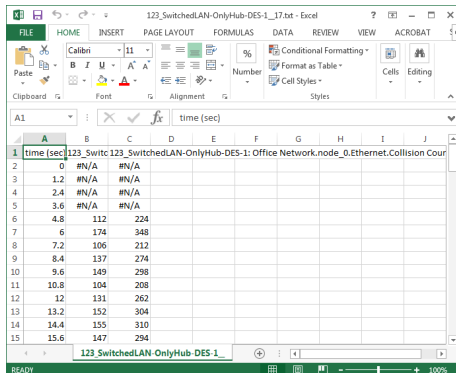
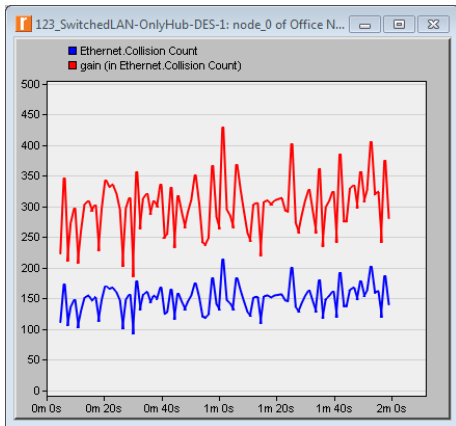
Predefined Filters (cont'd)

- **gain** – scales the ordinate values of a selected statistic by a fixed scale factor ('gain')
 - If gain = 1.0, then **gain = As Is**



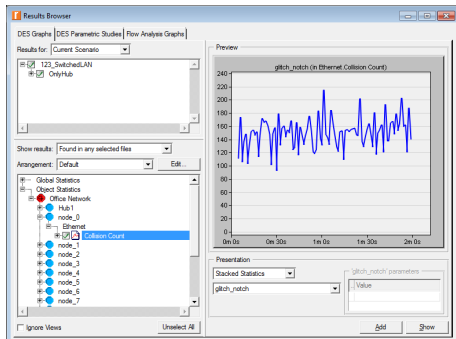
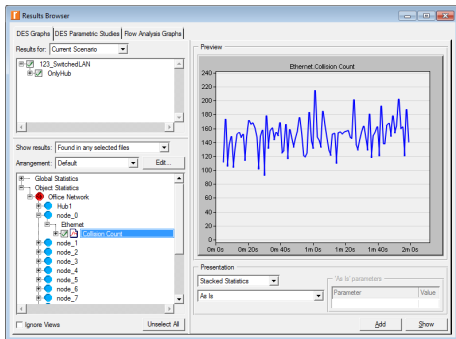
Predefined Filters (cont'd)

- $\text{gain}(i) = K \times Y_i$
 - E.g., $K = 2$



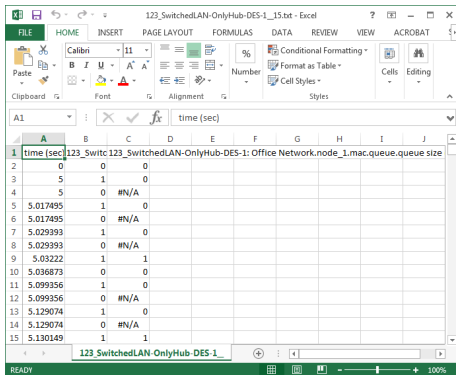
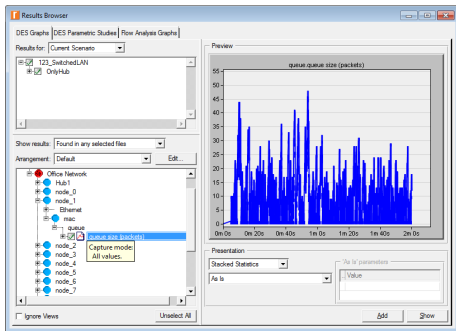
Predefined Filters (cont'd)

- **glitch_notch** – eliminates the occurrence of multiple entries that share the same abscissa value ('glitch')
 - If no glitches exist, then **glitch_notch** = **As Is**



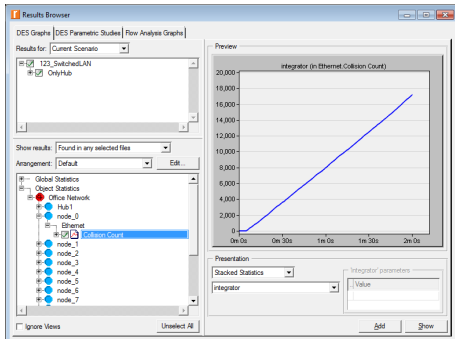
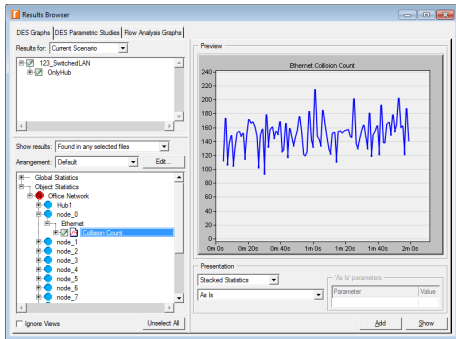
Predefined Filters (cont'd)

- Otherwise, it eliminates entries to ensure uniqueness of entries at each abscissa value



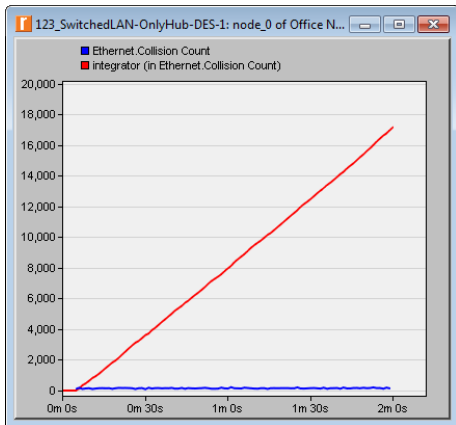
Predefined Filters (cont'd)

- **integrator** – displays the integral of a selected statistic with respect to its abscissa variable



Predefined Filters (cont'd)

- $\text{integrator}(n) = \sum_{i=0}^{n-1} Y_i \times (X_{i+1} - X_i)$

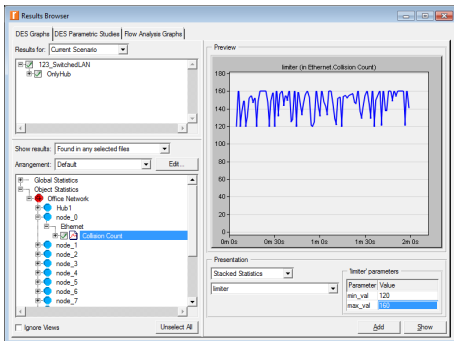
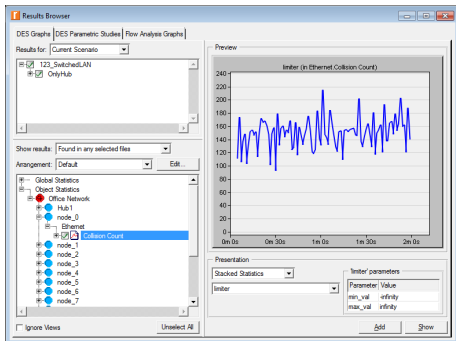


The Excel spreadsheet displays the data points for the graph. The columns are labeled 'time [sec]', 'Ethernet.Collision Count', and 'Integrator'. The data shows a linear increase in the integrator value over time, while the collision count remains near zero.

time [sec]	Ethernet.Collision Count	Integrator
0	#N/A	0
1.2	#N/A	0
2.4	#N/A	0
3.6	#N/A	0
4.8	112	0
6	174	134.4
7.2	106	343.2
8.4	137	470.4
9.6	149	634.8
10.8	104	813.6
12	131	938.4
13.2	152	1095.6
14.4	155	1278
15.6	147	1464

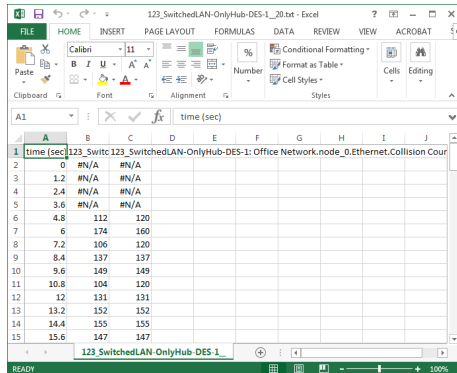
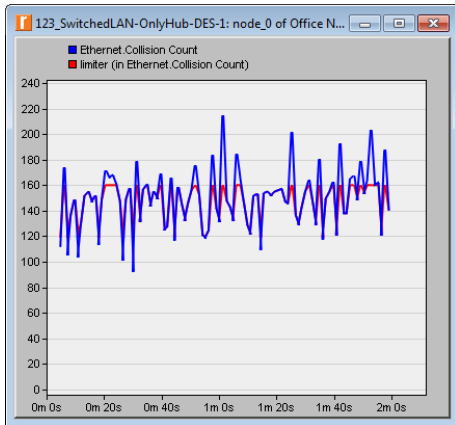
Predefined Filters (cont'd)

- **limiter** – constrains the ordinate values of a selected statistic within a specified range [min_val, max_val]
 - If $[-\infty, +\infty]$, then **limiter** = **As Is**



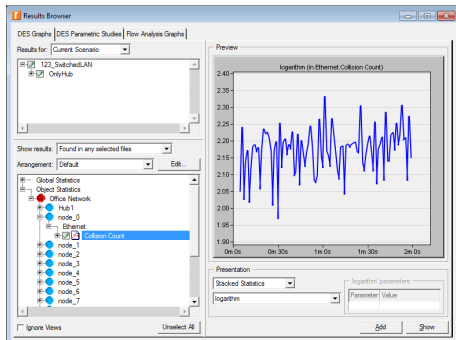
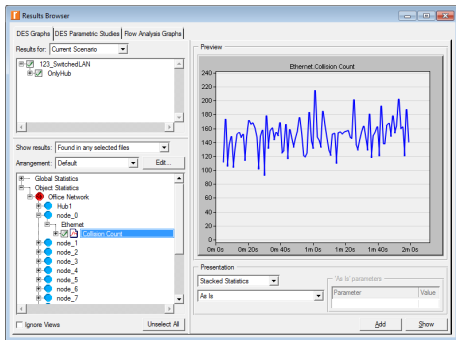
Predefined Filters (cont'd)

- $\text{limiter}(i) = \min\{\max\{\min_val, Y_i\}, \max_val\}$



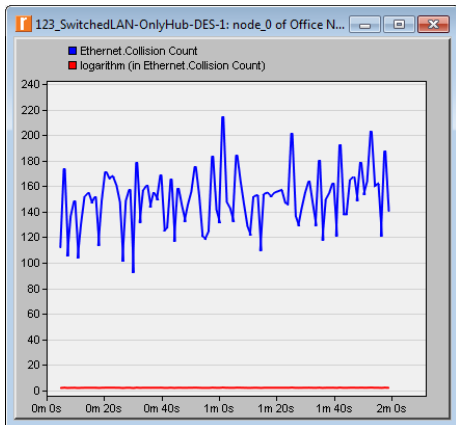
Predefined Filters (cont'd)

- **logarithm** – computes the base 10 logarithm of the ordinate entries of a selected statistic



Predefined Filters (cont'd)

- $\text{logarithm}(i) = \log_{10} Y_i$
 - If $Y_i = 0$, then $\text{logarithm}(i) = -\infty$
 - If $Y_i < 0$, then $\text{logarithm}(i) = \text{N/A}$

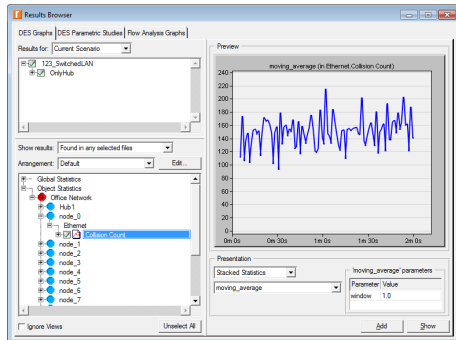
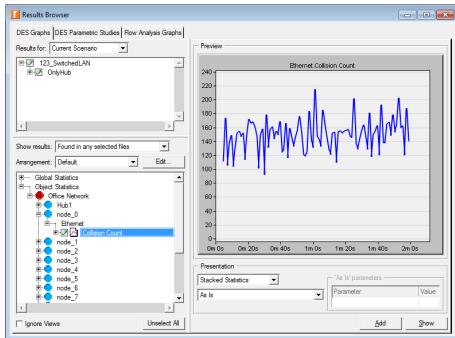


The screenshot shows an Excel spreadsheet with the following data:

time [sec]	123_Switic	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Collision Cour
0	#N/A	#N/A
1.2	#N/A	#N/A
2.4	#N/A	#N/A
3.6	#N/A	#N/A
4.8	112	2.049218
6	174	2.240549
7.2	106	2.025306
8.4	137	2.136721
9.6	149	2.173186
10.8	104	2.017033
12	131	2.117271
13.2	152	2.181844
14.4	155	2.190332
15.6	147	2.167317

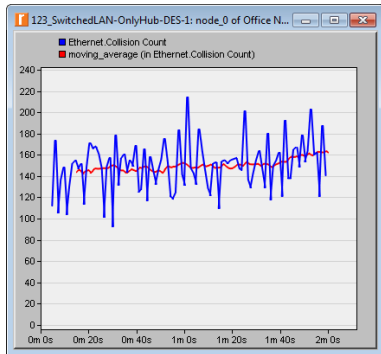
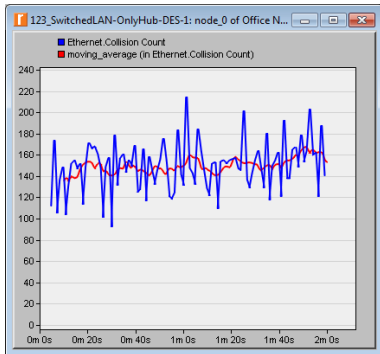
Predefined Filters (cont'd)

- **moving_average** – displays the continuous moving average of the ordinate values of a selected statistic over intervals of a specified width ('window')
 - If window $\approx dx$, then **moving_average** $\approx As Is$



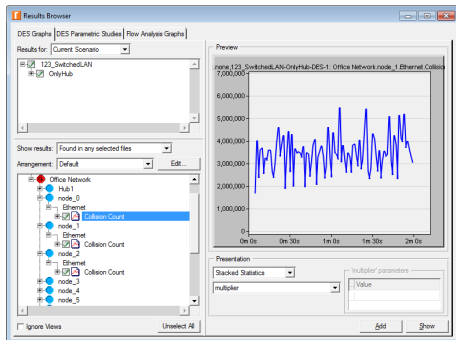
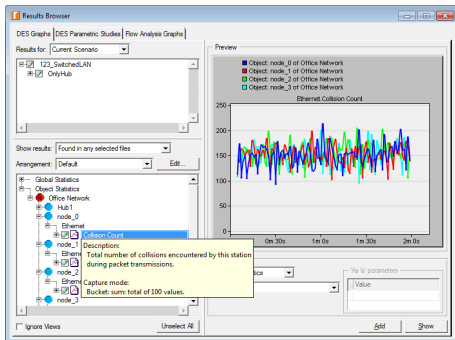
Predefined Filters (cont'd)

- When the 'window' parameter is set to a large value relative to the typical abscissa spacing between entries of the input statistic, the 'moving_average' filter provides a smooth result
 - E.g., **window = 10** vs. **window = 20**



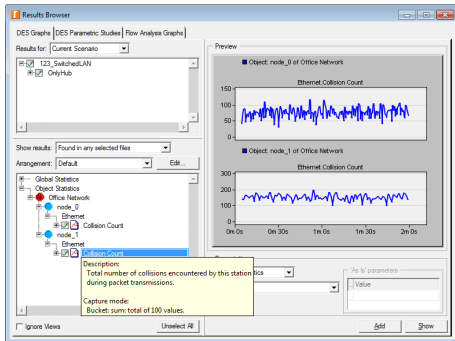
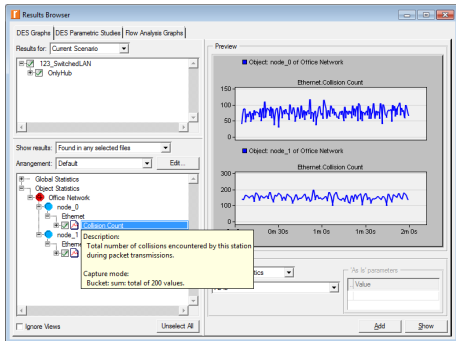
Predefined Filters (cont'd)

- **multiplier** – displays a graph that contains the product of values of one or more statistics
 - If the input statistics have exactly the same number of entries and these entries are aligned with respect to their abscissa values, then the output statistic can be computed simply by multiplying ordinate values for entries of equal abscissa



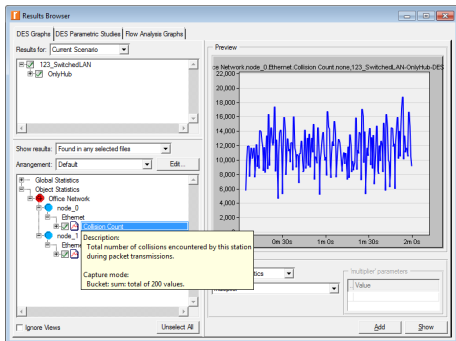
Predefined Filters (cont'd)

- What if the input statistics are not perfectly aligned with respect to each other?



Predefined Filters (cont'd)

- In such a case, an abscissa alignment mechanism is automatically applied by this filter before adding is performed

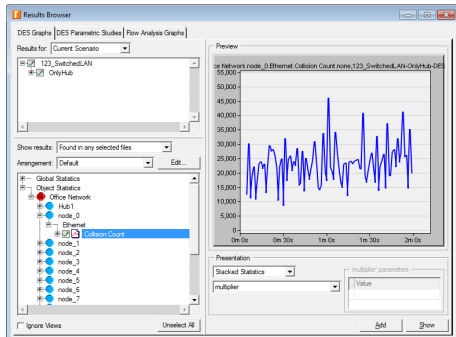
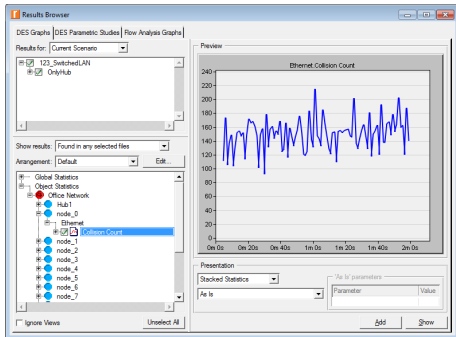


The screenshot shows an Excel spreadsheet with the following data:

time (sec)	123_Switic	123_Switic	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Col
0	#N/A	#N/A	#N/A
0.6	#N/A	#N/A	#N/A
1.2	#N/A	#N/A	#N/A
1.8	#N/A	#N/A	#N/A
2.4	#N/A	#N/A	#N/A
3	#N/A	#N/A	#N/A
3.6	#N/A	#N/A	#N/A
4.2	#N/A	#N/A	#N/A
4.8	42	139	5838
5.4	70	#N/A	9730
6	87	137	11919
6.6	87	#N/A	11919
7.2	47	164	7708
7.8	59	#N/A	9676

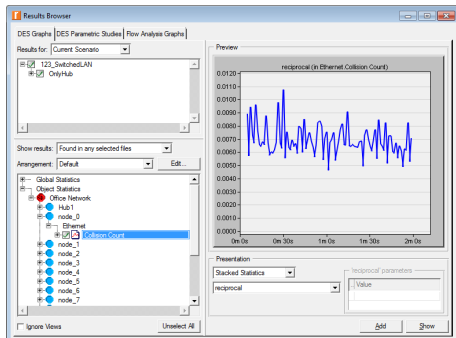
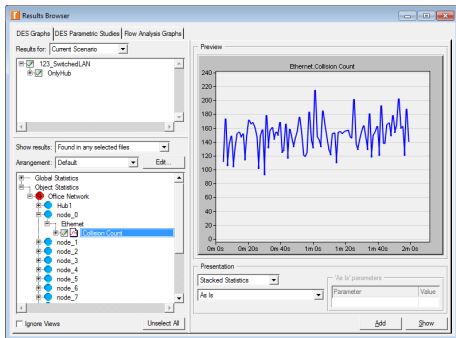
Predefined Filters (cont'd)

- In case of a single input statistic, $\text{multiplier}(i) = (Y_i)^2$



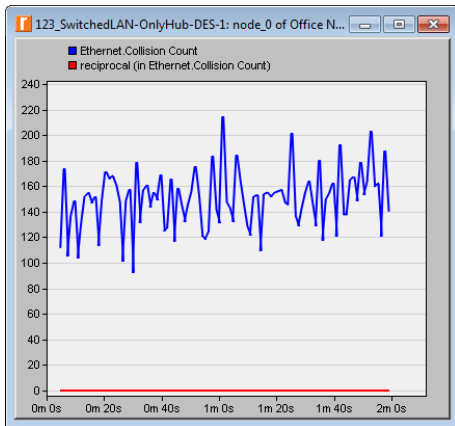
Predefined Filters (cont'd)

- **reciprocal** – displays the inverted ordinate values of a selected statistic



Predefined Filters (cont'd)

- $\text{reciprocal}(i) = \frac{1}{Y_i}$
 - If $Y_i = 0$, then $\text{reciprocal}(i) = \pm\infty$
 - If $Y_i = N/A$, then $\text{reciprocal}(i) = N/A$

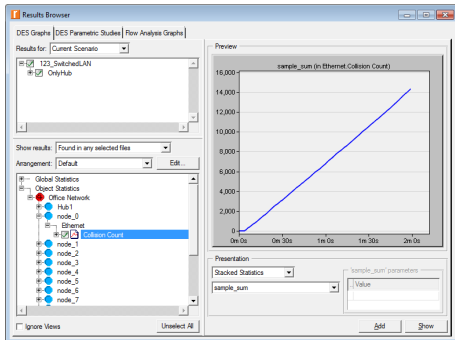
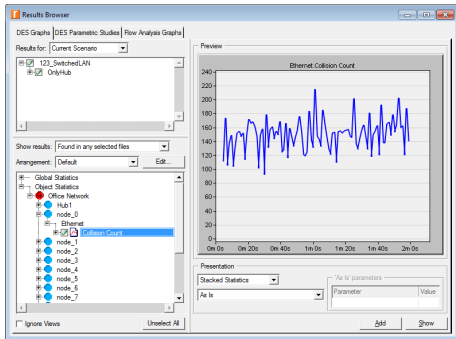


The screenshot shows an Excel spreadsheet with the following data:

time (sec)	123_Switic	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Collision Cour	reciprocal
0	#N/A	#N/A	#N/A
1.2	#N/A	#N/A	#N/A
2.4	#N/A	#N/A	#N/A
3.6	#N/A	#N/A	#N/A
4.8	112	0.008929	
6	174	0.005747	
7.2	106	0.009434	
8.4	137	0.007299	
9.6	149	0.006711	
10.8	104	0.009615	
12	131	0.007634	
13.2	152	0.006579	
14.4	155	0.006452	
15.6	147	0.006803	

Predefined Filters (cont'd)

- **sample_sum** – displays the running total of the ordinate values of a selected statistic



Predefined Filters (cont'd)

- $\text{sample_sum}(n) = \sum_{i=0}^n Y_i$

123_SwitchedLAN-OnlyHub-DES-1_11.txt - Excel

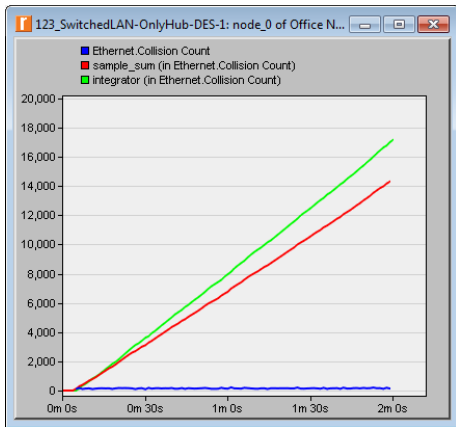
A	B	C
1	time (sec)	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Collision Cour
2	0	#N/A
3	1.2	#N/A
4	2.4	#N/A
5	3.6	#N/A
6	4.8	112
7	6	174
8	7.2	106
9	8.4	137
10	9.6	149
11	10.8	104
12	12	131
13	13.2	152
14	14.4	155
15	15.6	147

123_SwitchedLAN-OnlyHub-DES-1_10.txt - Excel

A	B	C
89	104.4	138
90	105.6	165
91	106.8	168
92	108	149
93	109.2	179
94	110.4	154
95	111.6	164
96	112.8	203
97	114	160
98	115.2	162
99	116.4	121
100	117.6	188
101	118.8	141
102	120	#N/A
103		14311

Predefined Filters (cont'd)

- If $dx \neq 1$, then **sample_sum** \neq **integrator**
 - If $dx = \text{const}$, then $\text{integrator}(n) = \text{sample_sum}(n) \times dx$

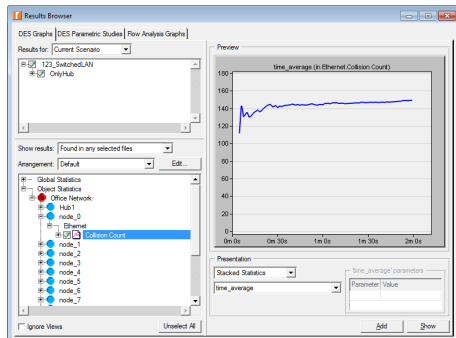
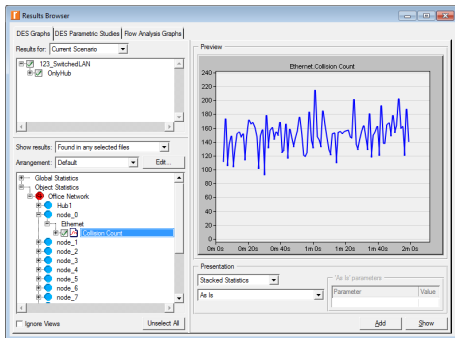


The screenshot shows an Excel spreadsheet with a table of data. The table has columns labeled A through J and rows numbered 89 to 103. The data in columns B through D appears to be time-related measurements.

	A	B	C	D	E	F	G	H	I	J
89		104.4	138	12357	14662.8					
90		105.6	165	12522	14828.4					
91		106.8	168	12690	15026.4					
92		108	149	12839	15228					
93		109.2	179	13018	15406.8					
94		110.4	154	13172	15621.6					
95		111.6	164	13336	15806.4					
96		112.8	203	13539	16003.2					
97		114	160	13699	16246.8					
98		115.2	162	13861	16438.8					
99		116.4	121	13982	16633.2					
100		117.6	188	14170	16778.4					
101		118.8	141	14311	17004					
102		120	#N/A	#N/A	17173.2					
103		120	#N/A	0	#N/A					

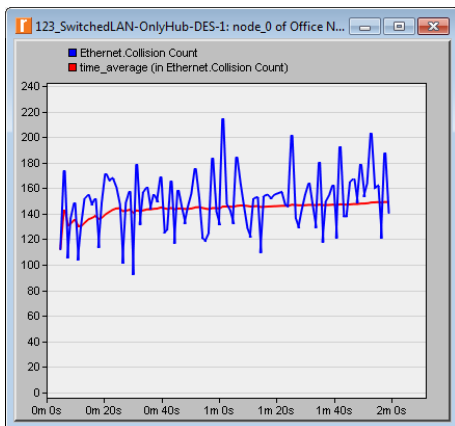
Predefined Filters (cont'd)

- **time_average** – displays the continuous moving average of the ordinate values of a selected statistic
 - The difference between 'time_average' and 'average' is that entry values are not weighted equally, but are instead weighted by the difference between their own abscissa and that of the subsequent entry



Predefined Filters (cont'd)

- $\text{time_average}(n) = \frac{\sum_{i=0}^{n-1} Y_i \times (X_{i+1} - X_i)}{\sum_{i=0}^{n-1} (X_{i+1} - X_i)}$
 - In effect, each value is weighted by the amount of time that the input statistic had that value



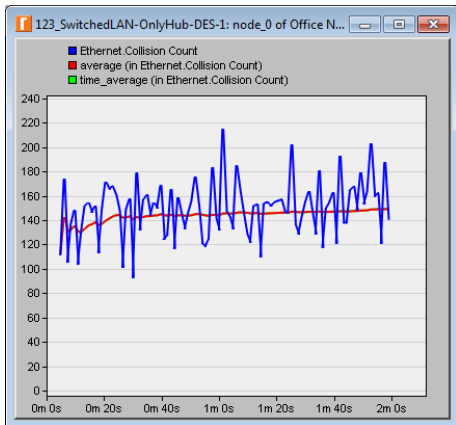
The Excel spreadsheet shows a table with the following data:

time (sec)	123_Switic	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Collision Cour
0	#N/A	#N/A
1.2	#N/A	#N/A
2.4	#N/A	#N/A
3.6	#N/A	#N/A
4.8	112	112
6	174	143
7.2	106	130.6667
8.4	137	132.25
9.6	149	135.6
10.8	104	130.3333
12	131	130.4286
13.2	152	133.125
14.4	155	135.5556
15.6	147	136.7

Predefined Filters (cont'd)

- If $dx = \text{const}$, then

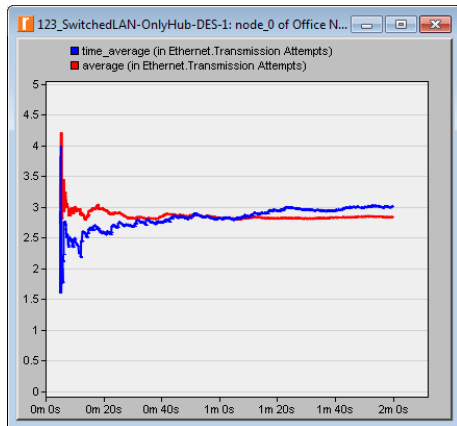
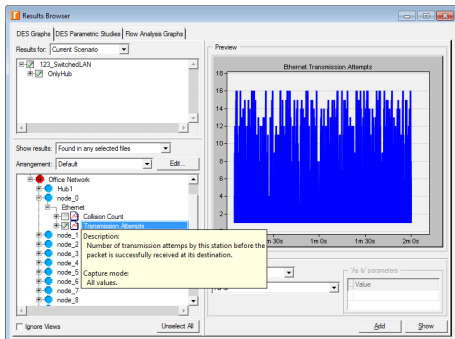
$$\text{time_average}(n) = \frac{\sum_{i=0}^{n-1} Y_i \times dx}{\sum_{i=0}^n dx} = \frac{dx \times \sum_{i=0}^{n-1} Y_i}{dx \times (n+1)} = \frac{\sum_{i=0}^{n-1} Y_i}{n+1} = \text{average}(n)$$



time (sec)	123_Swict	123_Swict	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Col
0	#N/A	#N/A	#N/A
1.2	#N/A	#N/A	#N/A
2.4	#N/A	#N/A	#N/A
3.6	#N/A	#N/A	#N/A
4.8	112	112	112
6	174	143	143
7.2	106	130.6667	130.6667
8.4	137	132.25	132.25
9.6	149	135.6	135.6
10.8	104	130.3333	130.3333
12	131	130.4286	130.4286
13.2	152	133.125	133.125
14	155	135.5556	135.5556
15.6	147	136.7	136.7

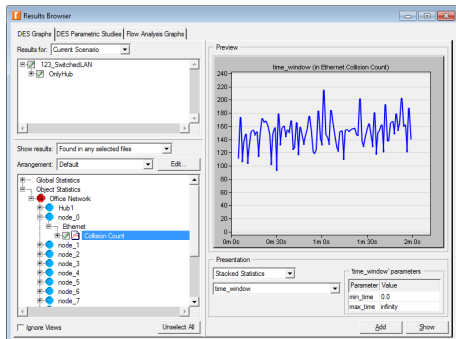
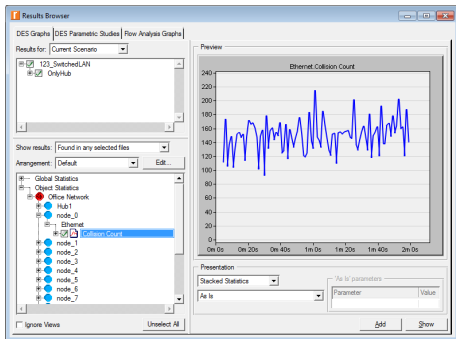
Predefined Filters (cont'd)

- If $dx \neq \text{const}$, then $\text{time_average} \neq \text{average}$



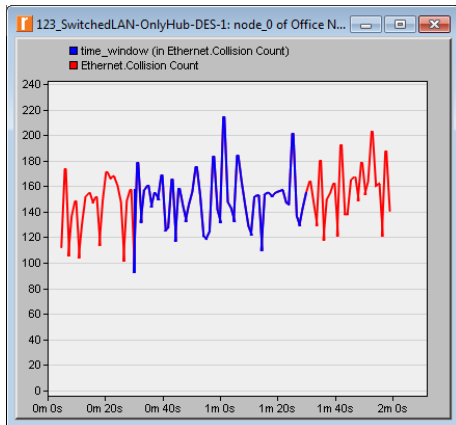
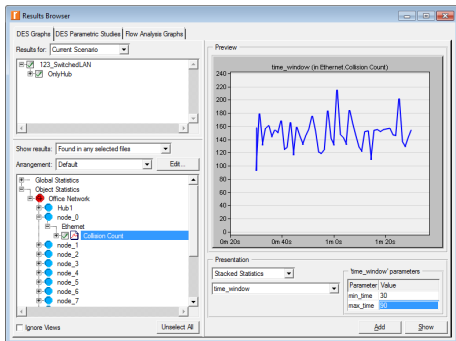
Predefined Filters (cont'd)

- **time_window** – eliminates all entries whose abscissa values lie outside a specified range
 - If $[0.0, +\infty]$, then **time_window = As Is**



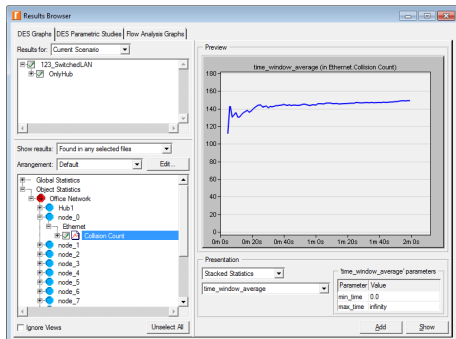
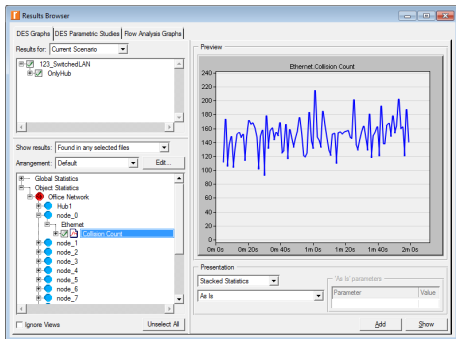
Predefined Filters (cont'd)

- The 'time_window' filter is equally applicable to any type of statistic, not only a time series



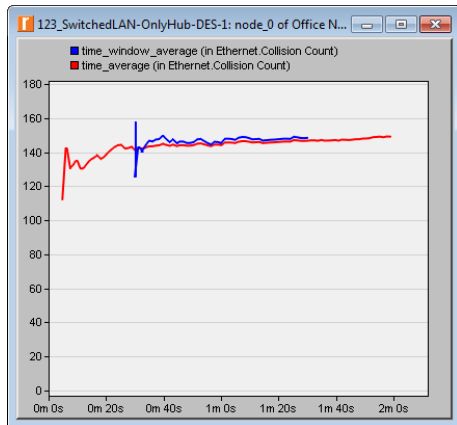
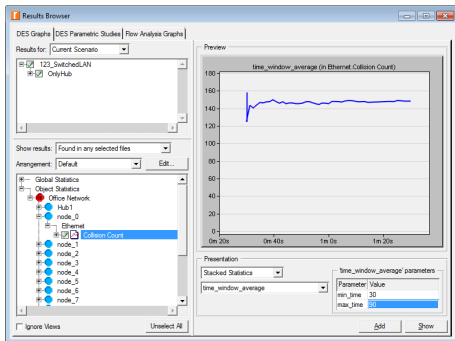
Predefined Filters (cont'd)

- **time_window_average** – displays the running continuous average of the ordinate values of a selected statistic within a specified range
 - If $[0.0, +\infty]$, then **time_window_average** = **time_average**



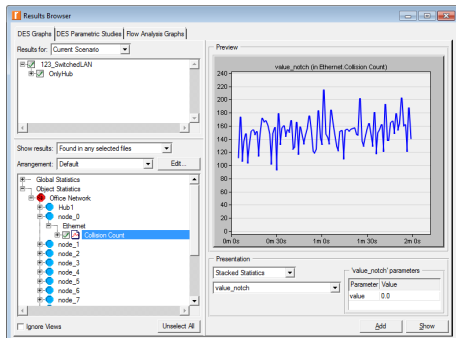
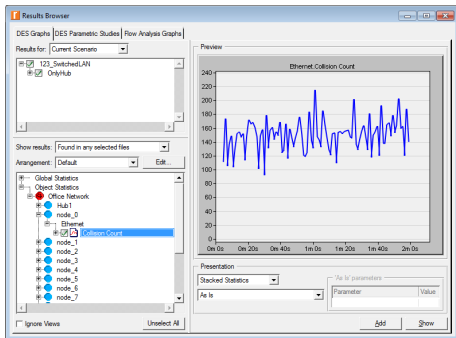
Predefined Filters (cont'd)

- Otherwise, **time_window_average** \neq **time_average**



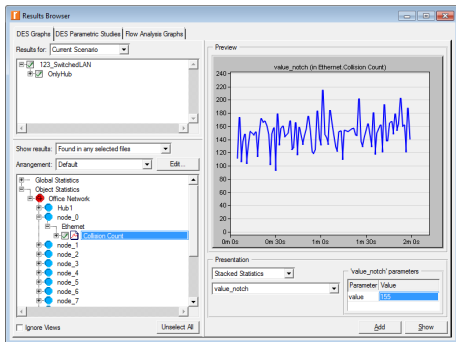
Predefined Filters (cont'd)

- **value_notch** – eliminates all entries that are equal to the specified value ('value')
 - If $value = 0.0$, then **value_notch** = **As Is**



Predefined Filters (cont'd)

- E.g., value = 155
 - Entries with this ordinate value are eliminated



time (sec)	123_Switch	123_SwitchedLAN-OnlyHub-DES-1: Office Network.node_0.Ethernet.Collision Cour
0	#N/A	#N/A
1.2	#N/A	#N/A
2.4	#N/A	#N/A
3.6	#N/A	#N/A
4.8	112	112
6	174	174
7.2	106	106
8.4	137	137
9.6	149	149
10.8	104	104
12	131	131
13.2	152	152
14.4	155	#N/A
15.6	147	147