

Identify Captured Resource Units

In addition to determining which resource you've captured, you can now find out the specific unit of that resource as well. With the new **OwnedResourceUnit()** function, you can get the unit number of any resource owned by an entity. This is useful when collecting custom resource statistics at the individual unit level.

In-Process Resource Utilization Statistics

Access the utilization of your resources at any time during simulation. Using the newly modified **PercentUtil()** function, you can find out the utilization of individual units of a resource or a summary of all units of a given resource type. This allows you to make dynamic logical decisions or write out custom statistics to an Excel file.

Utilization for a specific resource unit

```

LOGIC: ACTIVITY
1 Inc vStatsRow // Next row in Custom Stats array
2
3 // Log Resource Name
4 yCustomStats[vStatsRow, 1] = "Worker"
5 // Log Resource Unit
6 yCustomStats[vStatsRow, 2] = 5
7 // Log Resource Utilization
8 yCustomStats[vStatsRow, 3] = PercentUtil(Worker, 5)
  
```

Utilization for all units of a resource type

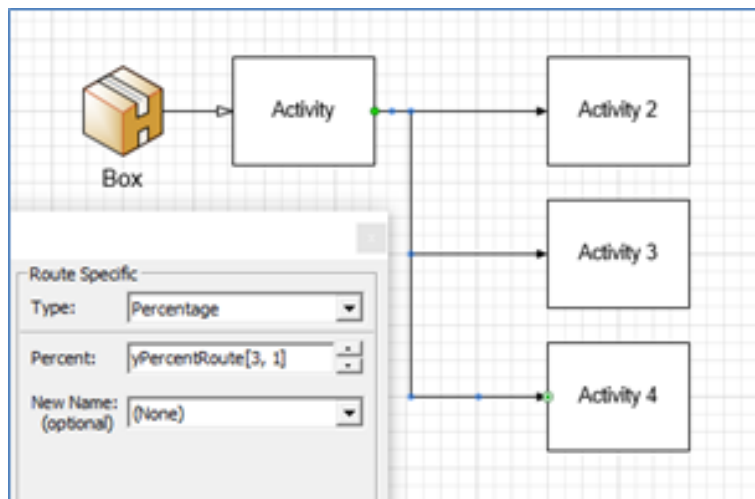
```

LOGIC: ACTIVITY
1 Inc vStatsRow // Next row in Custom Stats array
2
3 // Log Resource Name
4 yCustomStats[vStatsRow, 1] = "Worker"
5 // Log Resource Unit
6 yCustomStats[vStatsRow, 2] = "All"
7 // Log Resource Utilization
8 yCustomStats[vStatsRow, 3] = PercentUtil(Worker)
  
```

Initialize Routing Percentages from External Data

Percentage-based routes now have the flexibility to be initialized by external data stored in Excel using Arrays. This gives you the ability to fine-tune your model or do what-if analysis in Excel without the need to change the model.

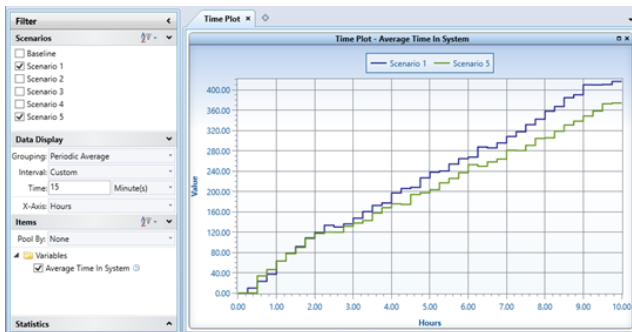
Utilize this capability previously only available through Macros and the Scenario Manager, by simply entering the Array in place of the percentage value in Routing properties.



Programmatic Export of Statistics

The statistical results of your simulation runs can be programmatically accessed through a new API to Output Viewer. You can get the raw data, down to the individual replication, or have it summarized or grouped (just like in Output Viewer) prior to accessing it. Either way, you can access your results, for example to load into Excel or a database, for analysis outside of Output Viewer.

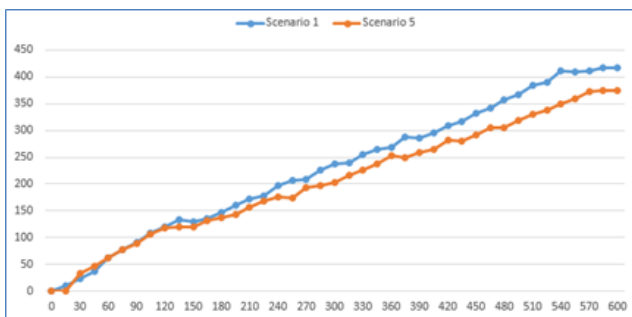
As an example, this is a Time Plot in Output Viewer where the time series data is averaged over a custom period of 15 minute intervals.



Using the new API, you can have Output Viewer summarize the data into 15 minute intervals prior to exporting it to Excel.

	A	B	C	D	E
1	Time Plot - Average Time In System (Avg. Reps)				
2					
3	Scenario	Replication	Element	Date/Time	Value
4	Scenario 1	Periodic Avg	Average Time In System	0	0.00
5	Scenario 1	Periodic Avg	Average Time In System	15	9.74
6	Scenario 1	Periodic Avg	Average Time In System	30	23.00
7	Scenario 1	Periodic Avg	Average Time In System	45	36.98
8	Scenario 1	Periodic Avg	Average Time In System	60	63.01
9	Scenario 1	Periodic Avg	Average Time In System	75	78.22
10	Scenario 1	Periodic Avg	Average Time In System	90	91.49
11	Scenario 1	Periodic Avg	Average Time In System	105	108.45

The exported format makes it easy to create a Pivot table and Pivot chart in Excel.



Enhancements

- When exporting Array data at the end of simulation, the replication number is no longer written out in the Excel Sheet name if **Export after final replication only** is checked.
- Minitab version 17.3 is now supported.