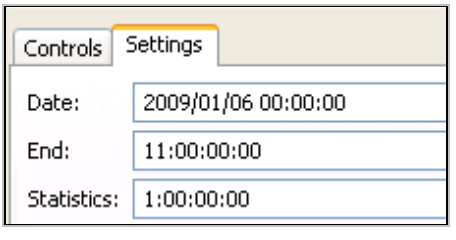
9 Statistics

9.1 Basics

Most questions in simulations deal with distributions of, for example, failures, waiting times, etc. For the evaluation of simulation runs, you can rely on statistical data, which the material flow objects collect.

9.1.1 Statistics Collection Period

The statistics collection period is the time interval between activating the collection of statistical data and the query of statistics. Statistics data are recorded only if the collection of statistics in the objects is active. If statistics is disabled, all statistical data of an object will be deleted. You can reset statistics collection in the EventController at a certain time. In this way, you can hide the ramp-up behavior of your model, and statistics collection can start when the system reaches full output. You can enter this setting in the EventController, on the tab Settings. Enter the time at which the event controller will reset statistics into the field **STATISTICS**.



With the setting above, the EventController will reset statistics of the objects after one day. When the simulation is finished after 11 days, the objects recorded statistical data for 10 days. The following description illustrates the composition of the statistics collection period (scheduled time only):

<table>
<tr><td colspan="6">Statistics collection period</td></tr>
<tr><td colspan="5">Resource not paused</td><td>R. paused</td></tr>
<tr><td colspan="3">Resource operational</td><td colspan="2">Resource not operational</td><td rowspan="2"></td></tr>
<tr><td>Waiting time</td><td>Set-up time</td><td>Working time</td><td>Blocked time</td><td>Failed time</td></tr>
</table>

Working time: A resource works when at least one MU is being processed on the object (setup times and recovery times are not included in the working time).

S. Bangsow: Manufacturing Simulation with Plant Simulation, Simtalk, pp. 223 – 252, 2010.

Failed time: A resource is failed, if it is not paused and its attribute failed has the value True.

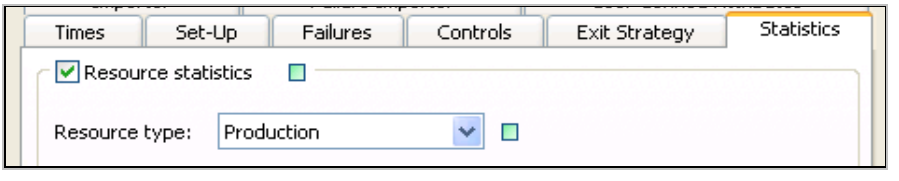
Blocking time: A resource is blocked if it

- is full
- neither failed nor paused and
- all places do not work (e.g., the MU is processed and cannot be passed on).

Paused: A resource is paused when its paused attribute has the value True. In addition, more than 100 different values for statistical analysis are available.

9.1.2 Activating Statistics Collection

Open a material flow object (double-click) – click the tab Statistics. Enable Resource statistics, and click OK or Apply.



You can also enable or disable statistics collection of an object using the method

```
<path>.ResStatOn:=true; --or false
```

Note:
By default, statistics collection is turned on for all material objects. To increase the performance of the simulation, it can help to deactivate statistics collection for all objects for which you do not need statistical analysis.

9.2 Statistics – Methods and Attributes

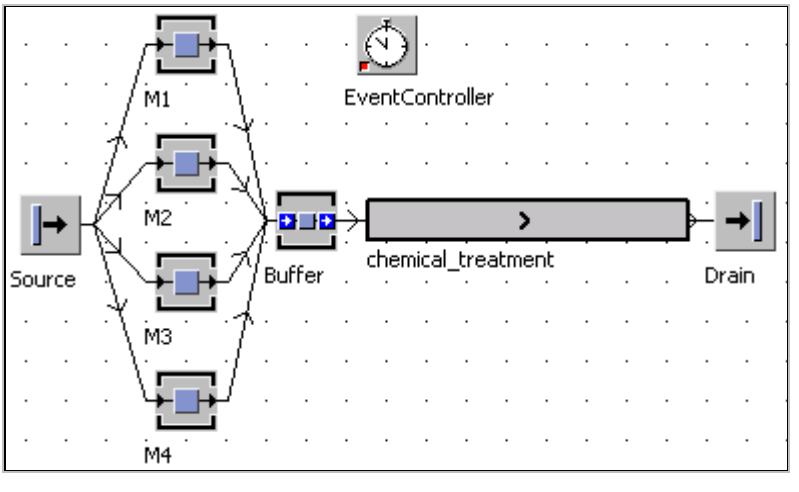
You can read all statistics data with SimTalk.

Method	*Description*
`<path>.statistics`	Shows statistics of the object on screen
`<path>.statistics(<table>)`	Statistics will be written in the specified Plant Simulation table.
`<path>.statistics(<string>)`	Statistics will be written in the specified file.
`<path>.statWaitingPortion`	Returns the percentage of the waiting time relative to the total time (data type real).

<path>.statWorkingPortion <path>.statBlockingPortion <path>.statFailPortion <path>.statSetupPortion <path>.statPausingPortion <path>.statUnplannedPortion <path>.statEmptyPortion	see above
<path>.statNumIn <path>.statNumOut	Returns the number of MUs that entered (leaved) the object. The returned data type is integer.
<path>.statMaxNumMU	Returns the maximum number of places occupied during the simulation.
<path>.initStat	The method initStat resets the statistics of an object. This can be useful if the statistics recording should not start at the beginning of the simulation, the statistics collection will contain only values from the call of initStat.

Example 96: Statistics

You are to simulate a manufacturing cell. Four machines feed a chemical treatment unit. The chemical treatment unit has a constant feed rate of 0.00333 m/s. The part has a length of 0.4 m. The machines have a processing time of 4 minutes, an availability of 50% and a MTTR of 3 hours. A buffer with capacity of 100 parts is located in front of the chemical treatment unit, which is 30 meters long. The source produces parts at an interval of 2 minutes. Create the following Frame:



This example will show some typical statistical analysis.

1. The statistical data is to be written into a file at the end of the simulation. This can be easily accomplished with a table. During or at the end of a simulation, you write the statistics data into a table and then save the table as a file. For this the TableFile provides the following methods:

Method	*Description*
`<path>.writeFile(<string>)`	`This method writes the contents of the table into a text file. Pass the path as argument. Existing files with the same name will be overwritten.`
`<path>.writeExcelFile( <string>,[<string>])`	`This method writes the contents of the table into an Excel file. Pass the filename (path). As a second argument you can pass an Excel table name. To be able to use this method, MS Excel has to be installed on your computer`

To simplify statistical analysis, write a method, which writes the statistical data of all material flow objects to a TableFile.

Add a TableFile "analysis" to the frame. Format the table according to the following example:

	string 0	real 1	real 2	real 3	real 4	real 5
string		working	waiting	blocked	failed	paused
1						
2						
3						

Preliminaries: With the help of the frame object, you can access all objects in the frame by an index (method: <path>.node(<integer>)). The method <path>.numNodes returns the number of objects within the frame. Finally, you can query the class of the objects with <path>.class (e.g., MateriaFlow.SingleProc).

Add an endSim-method to the frame. The method iterates through all objects in the frame. For objects of class SingleProc, the method inserts a row into the table and writes its name and statistical values into it. Finally, the method exports the contents of the TableFile to an Excel file: simulation_analysis.xls.

Method endSim:

```
-- writes statistical data for all SingleProc
-- objects
is
  i:integer;
  obj:object;
```

```
do
  analysis.delete;
  for i:=1 to current.numNodes loop
    obj:=current.node(i);
    if obj.class = .MaterialFlow.SingleProc then
      current.analysis.writeRow(0,
      current.analysis.YDim+1,
      obj.name,obj.statWorkingPortion,
      obj.statWaitingPortion,
      obj.statBlockingPortion,
      obj.statFailPortion,
      obj.statPausingPortion);
    end;
  next;
  -- write  excel file
  current.analysis.writeExcelFile(
  "c:\simulation_analysis.xls");
end;
```

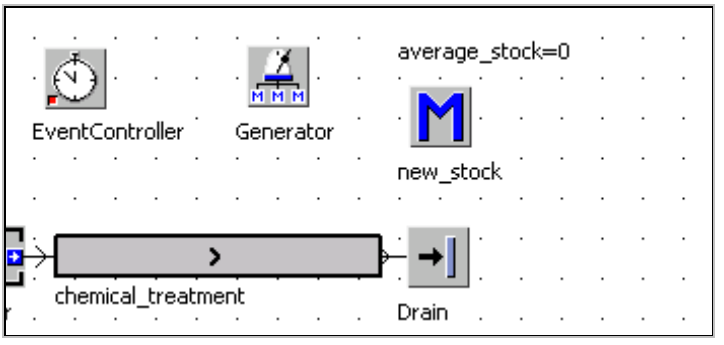
You can easily extend this method by adding other classes. Set a simulation end of 2 days, and let the simulation run up to this point. Search for the Excel file on your "c:\" drive.

	A	B	C	D	E	F
1		working	waiting	blocked	failed	paused
2	M1	0,28	0,02	0,16	0,54	0,00
3	M2	0,33	0,03	0,12	0,52	0,00
4	M3	0,29	0,03	0,14	0,54	0,00
5	M4	0,24	0,02	0,19	0,56	0,00

2. Determining Average Values

Often average values need to be calculated within a simulation. A typical example is the calculation of average stock. For an average calculation, you need a series of values and the number of the values (arithmetic mean). Within the simulation you can choose another approach. A generator calls a method every hour that determines the number of parts within the frame. The method calculates a new average based on the old average, the number of hours, and the new stock. You can easily find out the number of MUs in the frame with <path>.NumMu. (Keep in mind though that the method numMU also counts containers and transporters.)

Add the variable average_stock (real) to the frame. Insert a generator into the frame and a method "new_stock". The generator calls "new_stock" once per hour starting after 1 hour.



Program the method new_stock:

```
is
  hours:integer;
do
  hours:=time_to_num(eventController.simTime)/3600;
  average_stock:=
  (average_stock*(hours-1)+ current.numMu)/hours;
end;
```

3. Record Values

For the evaluation and optimization of the simulation, you often need the progression of values over time. You can accomplish this in two ways:

a) Record values with TimeSequence objects (a separate TimeSequence for each value).
b) Record values with a table and analysis of the table.

Example: In the example above, you are to show the distribution of the failures of the individual machines (hourly). If a failure has occurred, the value 1 should be entered, if the object is not failed at the moment, the value 0 should be entered. Insert the table failure_machines into the frame and format it as follows:

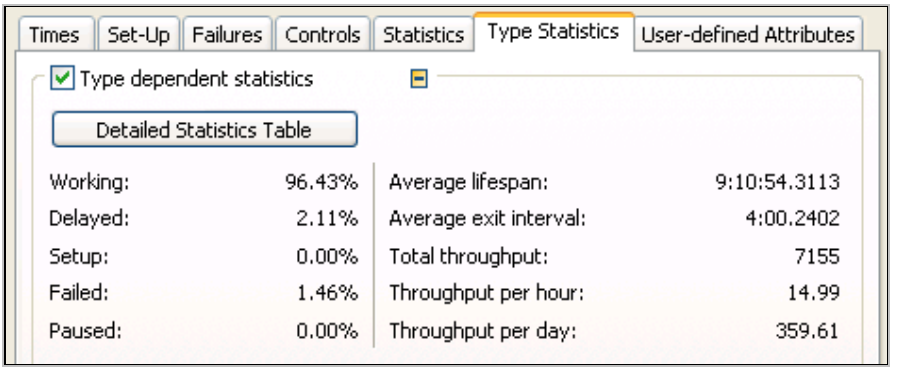
	time 1	integer 2	integer 3	integer 4	integer 5
string	time	M1_failure	M2_failure	M3_failure	M4_failure
1					
2					
3					

Program the method record_failures:

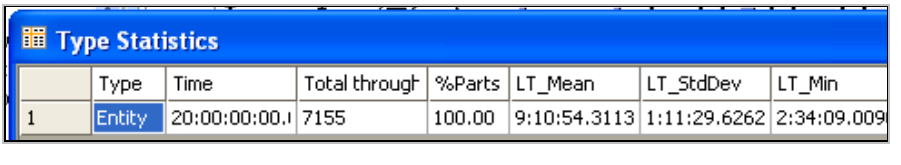
```
is
  i:integer; -- next entry
do
  i:=failure_machines.YDim+1;
  -- call once per hour
  failure_machines[1,i]:=eventController.simTime;
  -- set values depending of attribute failed
  if M1.failed then
    failure_machines["M1_failure",i]:=1;
  else
    failure_machines["M1_failure",i]:=0;
  end;
  if M2.failed then
    failure_machines["M2_failure",i]:=1;
  else
   failure_machines["M2_failure",i]:=0;
  end;
  -- and so on
end;
```

4. Data Collected by the Drain

The drain collects detailed statistics about the destroyed parts. Open the drain, and select the tab: Type Statistics.



Click the button Detailed Statistics Table to receive further information.



The drain provides a number of methods for accessing statistics; here is a small selection:

Method	*Description*
`<path>.typeStatistics(<table>)`	Copies the type statistics table in the specified table
`<path>.typeStatisticsCumulated (<table>)`	Analogous
`<path>.statThroughputPerHour`	Returns the throughput per hour (real)
`<path>.ThroughputPerDay`	Returns the throughput per day
`<path>.statAvgLifeSpan`	Returns the average throughput time

Most statistical data can be understood more easily if they are presented graphically. For this purpose, Plant Simulation provides user interface objects.

9.3 User Interface Objects

9.3.1 Chart

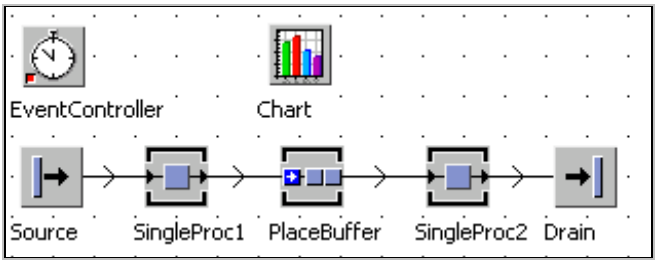
The object Chart represents data in Plant Simulation graphically. In watch mode, the graphic is automatically updated after each modification of a displayed value. (The value must be observable.) This way you can visualize the dynamic behavior of certain values during the simulation.

9.3.1.1 Plotter

In the following example, the development of stock in a buffer is to be presented graphically.

Example 97: Plotter

Create the following Frame:



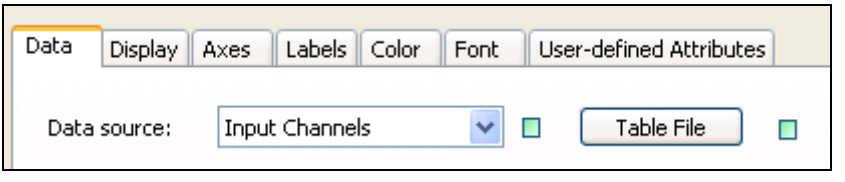
Settings: SingleProc1, SingleProc2 processing time 1 minute each, availability SingleProc1 and SingleProc2 95% MTTR 3 hours, use different random streams, PlaceBuffer capacity 100, accumulating, 30 seconds processing time.

There are two ways to present data in charts: From input channels or from TableFiles.

Input Channel
The Chart object itself records and displays the data. You can access the recorded values using SimTalk.

Example: You are to display the stock in the PlaceBuffer.

*1. Click the tab Data in the Chart and select the **DATA SOURCE – INPUT CHANNELS**.*

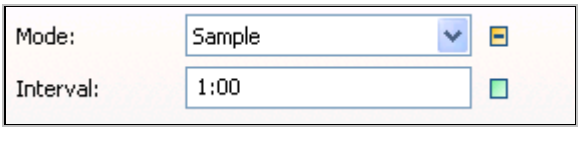


*Clicking the button opens a table into which you can enter the name of the object, the path of the displayed value and comments. First, turn off inheritance of the table (+ Apply). Then click the button **TABLE FILE**. Enter the path and the attribute, which is to be displayed (PlaceBuffer.numMU in our example.)*

	string 0	string 1	string 2
string		stock	
1		PlaceBuffer.numMU	
2			

You still need to select at which interval Plant Simulation is to update the chart on the tab Data. Watch mode updates after every change of the observable value, sample mode updates within the set interval, and plot mode updates the graph after each simulation event.

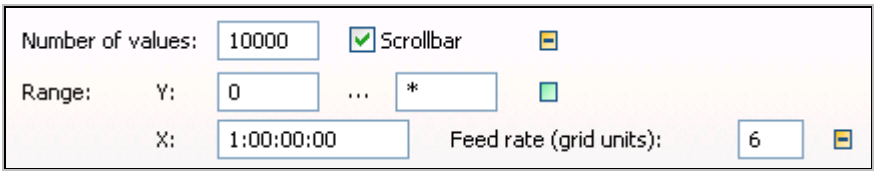
Example: The chart will be updated every minute, the setting for this is as follows:



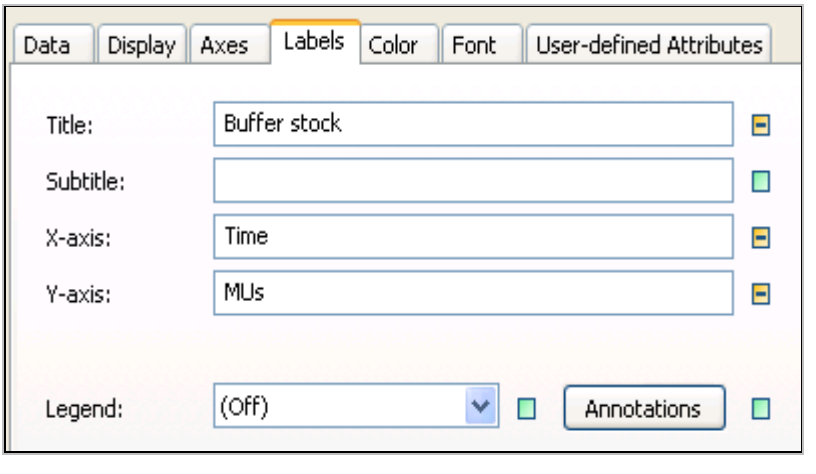
2. Select the category plotter and Chart type line on the tab Display.

If you select the option Display in frame, Plant Simulation shows the diagram (with its values) instead of the object's icon in the frame.

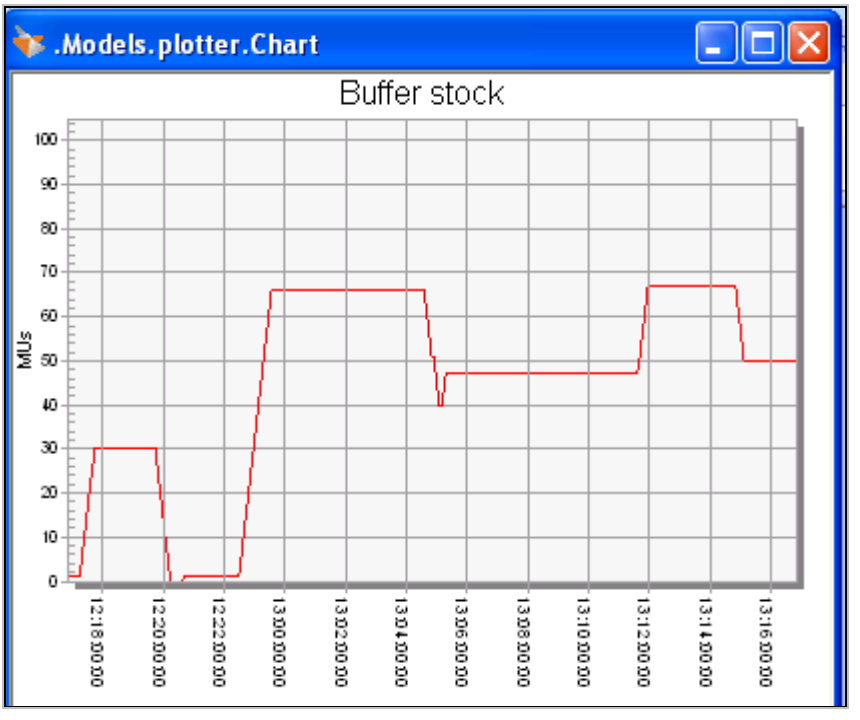
3. You then have to select some settings on the tab ***AXES****. Enter the number of displayed values (e.g., 10,000). The scroll bar option is mostly useful for presentations using a plotter. In addition, you have to enter the size of the displayed time range in the plotter window in the box* ***RANGE X****, enter 1:00:00:00 (one day).*



4. Add labels to the plotter on the tab ***LABELS****. Here you can also specify whether a legend is displayed and how it is displayed in the chart window.*



You can select additional format settings on the tabs ***FONT AND COLOR****. Clicking the button* ***SHOW CHART*** *opens the window of the plotter:*



You can access the data, which the plotter records, and write them into a TableFile.

Example: Add a TableFile (analysis) and a method (saveData) to the frame. You can read the data of the chart object with the method:

```
<path>.putValuesIntoTable(<table>)
```

First, turn off inheritance for the TableFile.

Program the method saveData:

```
is
do
  -- delete previous values
  analysis.delete;
  -- write the chart data into the table analysis
  chart.putValuesIntoTable(analysis);
end;
```

Plant Simulation formats the table and inserts the data of the plotter.

9.3.1.2 Chart Types

Plant Simulation provides a large number of different chart types to display values.

Example 98: Chart from a TableFile

Continuing the example above:

You are to show the composition of the statistics collection period of the objects SingleProc1 and SingleProc2. You want to write the values into a TableFile. You want to show the values of the table in a chart. The following values will be displayed:

Value	*SimTalk Attribute*
Waiting time	<path>.statWaitingPortion
Working time	<path>.statWorkingPortion
Blocked time	<path>.statBlockingPortion
Failed time	<path>.statFailPortion
Paused time	<path>.statPausingPortion

Insert a TableFile with the following formatting (chart_values) into the Frame:

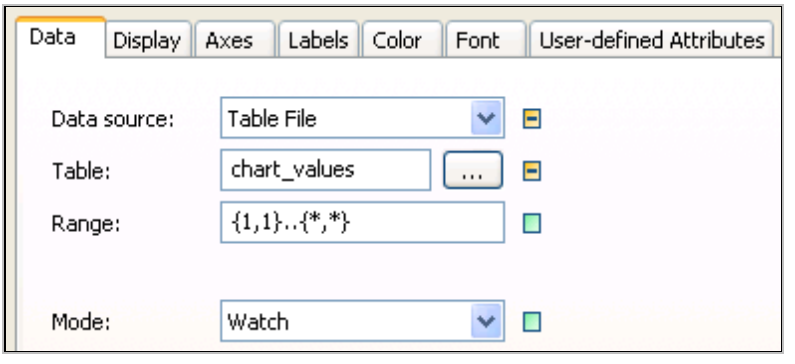
	string 0	real 1	real 2	real 3	real 4	real 5
string		waiting	working	blocked	failed	paused
1	SingleProc1					
2	SingleProc2					

At the end of the simulation run, a method (endSim) will write the statistics values of the two SingleProc into the TableFile chart_values.
Program the method endSim:

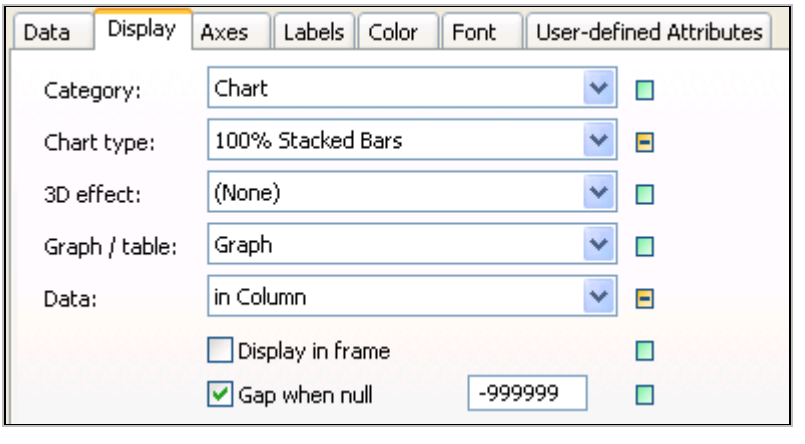
```
is
do
  -- values of SingleProc1
  chart_values.writeRow(1,1,
    SingleProc1.statWaitingPortion,
    SingleProc1.statWorkingPortion,
    SingleProc1.statBlockingPortion,
    SingleProc1.statFailPortion,
    SingleProc1.statPausingPortion);
  -- values of SingleProc2, next row
  chart_values.writeRow(1,2,
    SingleProc2.statWaitingPortion,
    SingleProc2.statWorkingPortion,
    SingleProc2.statBlockingPortion,
    SingleProc2.statFailPortion,
    SingleProc2.statPausingPortion);
end;
```

Creating Charts

1. Insert a Chart object into the frame. Open the chart and select the Data source – TableFile on the tab ***DATA****. Enter the name of our table, chart_values, into the field* ***TABLE****:*



2. Select ***CATEGORY*** *>* ***CHART*** *on the tab* ***DISPLAY****. The chart-type stacked bars (100%) is suited for displaying the portions of the statistics collection period. You need to set* ***DATA – IN COLUMN*** *(structures of the table chart_values – the data are spread across several columns).*



3. Label your chart. Show the legend on tab ***LABELS****. To avoid misunderstandings, you must customize the colors in the chart so that they match the colors of the status LEDs (in any case, failures should be shown red, pauses blue, and blockages orange). You can set the order of colors on the tab* ***COLOR****. Double-click a color to change it, and then select a new color. The order of colors in the example above must be yellow, green, orange, red, and blue.*

The chart will be shown by clicking the button Show chart. You can also show it by using the context menu of the chart object icon.

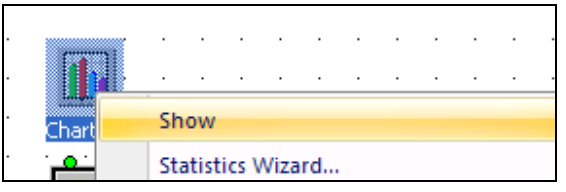
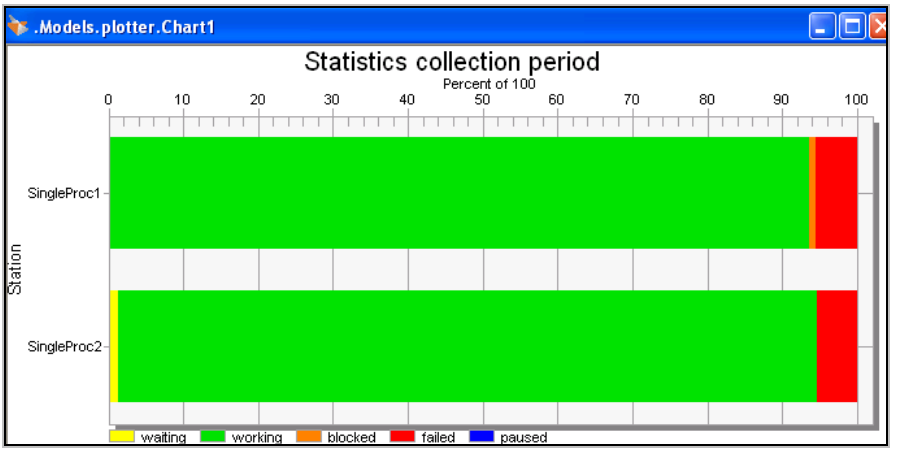


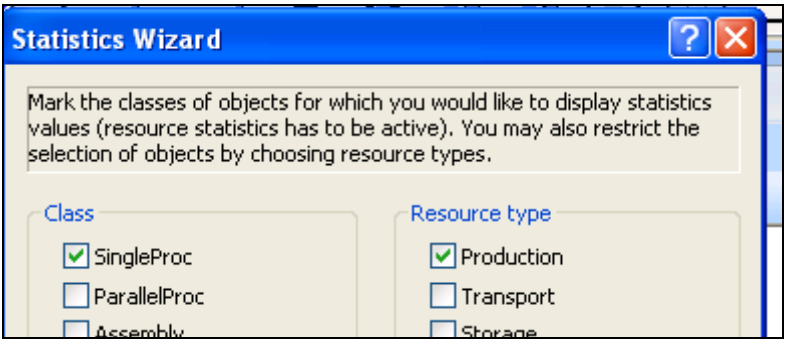
Chart type-Stacked bars:



9.3.1.3 Statistics Wizard

If you want to analyze the statistics collection period of all objects of a certain class, you can use the statistics wizard. For that, add a chart object to the frame. Click the right mouse button on the chart icon in the frame. Select **STATISTICS WIZARD** from the context menu.

Select the objects whose statistics you want to show in the dialog of the statistics wizard. Leave SingleProc and Production checked.



9.3.1.4 Histograms

Histograms show the frequency of certain values in relation to the simulation time.

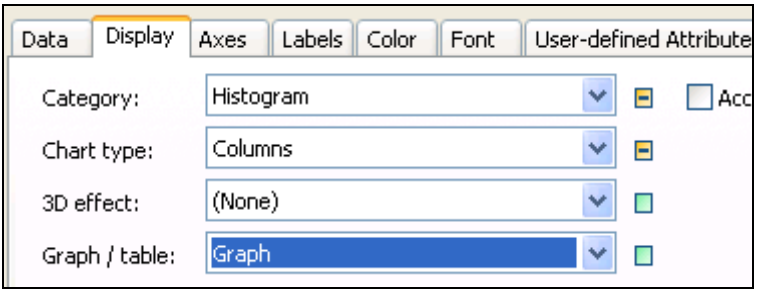
Example 99: Histogram

We continue with the example above: You are to display the distribution of the occupancy of the PlaceBuffer (attribute numMU).

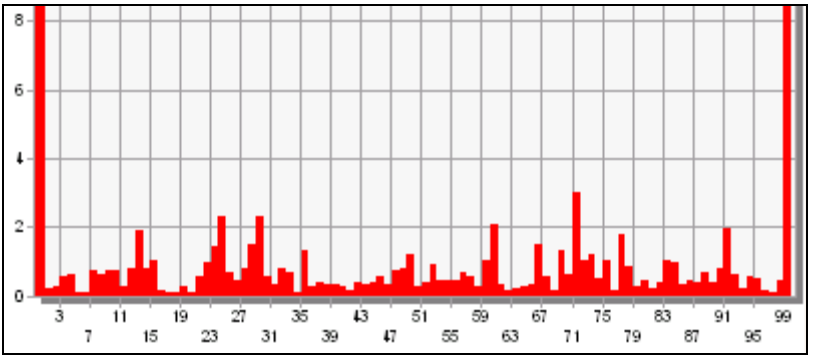
1. Add a new chart to the frame. Select ***DATA – INPUT CHANNELS*** *and enter PlaceBuffer.numMU in the table.*

	string 0	string 1	string 2
string		Stock	
1		PlaceBuffer.numMU	

2. Select ***CATEGORY – HISTOGRAM*** *and the* ***CHART TYPE – COLUMNS*** *on the tab* ***DISPLAY****.*



*3. Clicking **SHOW CHART** displays the histogram.*



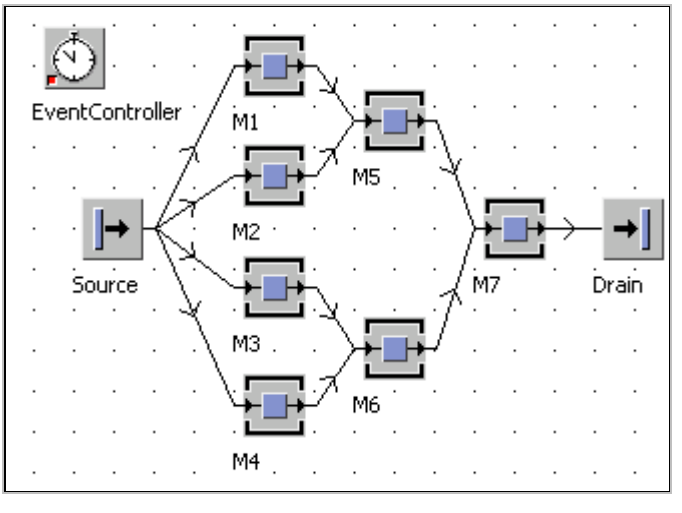
Note:
You can copy charts in the dialog box of the chart object to the clipboard and then paste them into other programs (e.g., PowerPoint). Select Tools – Copy to Clipboard in the dialog of the Chart.

9.3.2 The Sankey Diagram

The Sankey diagram is used for visualizing the distribution of the material flow. For this, Plant Simulation uses lines with different widths. The Sankey diagram is located in the folder Tools, or on the toolbar Tools.

Example 100: Sankey Diagram

The following frame shows how the Sankey diagram works:

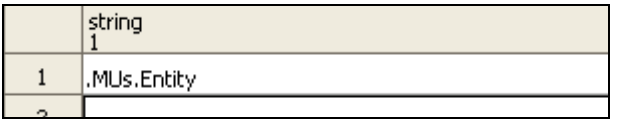


Settings:

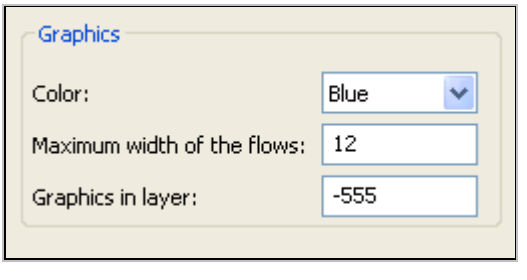
Machine	*Processing time*	*Availability*	*MTTR*
M1	*1:00.0000*	*95%*	*2:00:00.0000*
M2	*1:00.0000*	*85%*	*2:00:00.0000*
M3	*1:00.0000*	*70%*	*2:00:00.0000*
M4	*1:00.0000*	*50%*	*2:00:00.0000*
M5	*1:00.0000*	*95%*	*2:00:00.0000*
M6	*1:00.0000*	*85%*	*2:00:00.0000*
M7	*50.0000*	*95%*	*2:00:00.0000*

The source produces parts with an interval of 1 minute (blocking), the exit strategy is cyclic blocking. Add a SankeyDiagram to the frame. Open the SankeyDiagram by double-clicking it. Click the button Open (MUs to be watched).

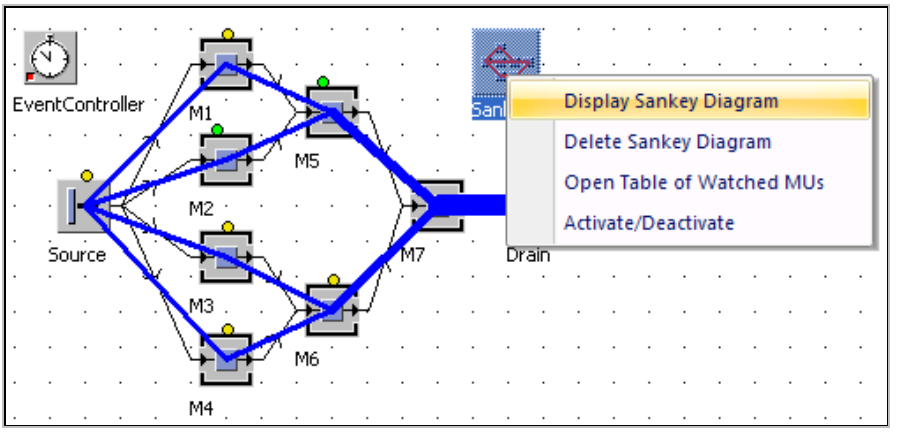
Enter the MU class, which is to be observed, into the following table. Drag the class Entity from the class library into the table.



You can select some formatting options, such as color settings and the maximum width of the streams:

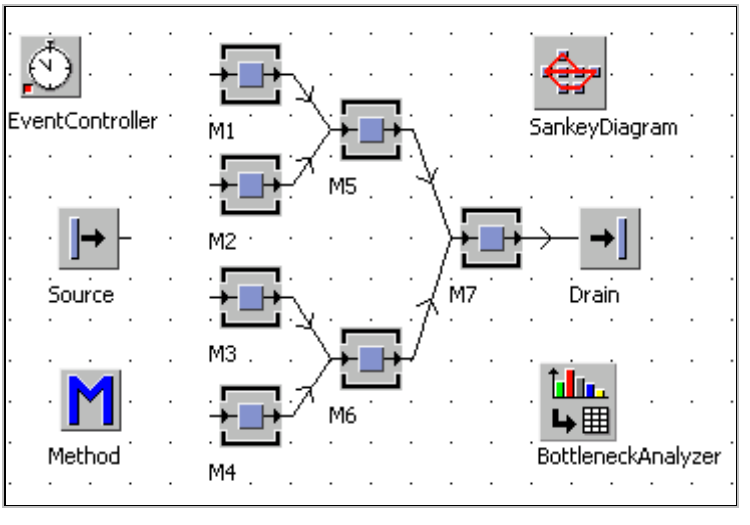


Graphics in layer determines the z-position of the Sankey display. The smaller the number, the closer to the foreground a graphic is located. Finish your settings by clicking OK. Now, run the simulation for a while (50 days). Then click the right mouse button on the object SankeyDiagram in your frame. Select ***DISPLAY SANKEY DIAGRAM****.* ***DELETE SANKEY DIAGRAM*** *deletes the Sankey streams.*



The thicker the Sankey streams between two stations, the more MUs have been transported on the connectors or methods between these stations. The exit strategy cycle of the source leads to the stations M1 to M4 receiving the same number of parts. If a machine fails, the source waits with the transfer process until the machine is operational again. M1 to M4 receive the same number of parts (Sankey lines have the same width). Output after 50 days is 21,230 parts.

You are to simulate a second variant. Click the right mouse button on the Frame in the class library, and select Duplicate. Close the frame window, and open the duplicate. Change this frame as follows:

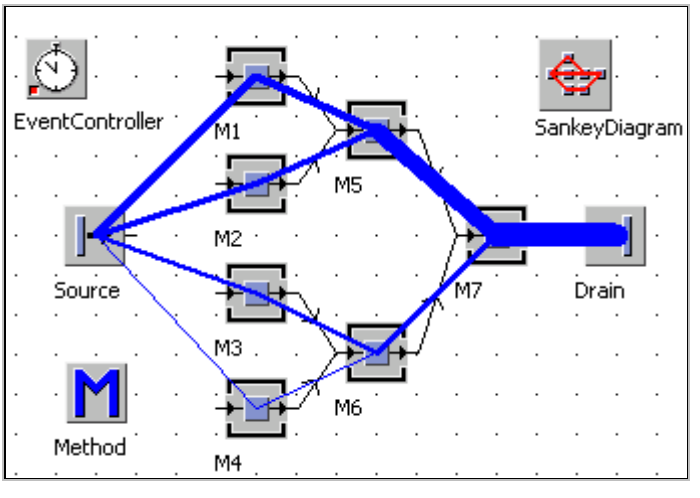


Program the method and assign it as the exit control (front) to the source. The source is to transfer the parts to the first available and operational machine.

Method:

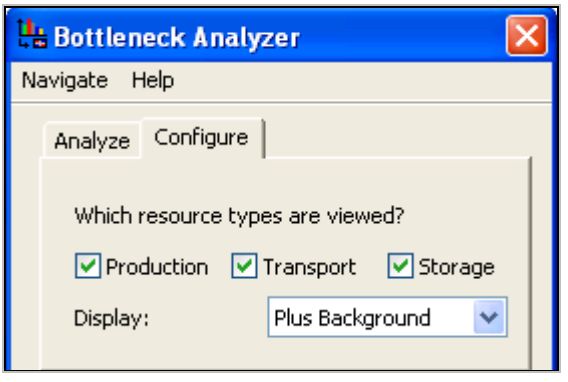
```
is
do
  waituntil (m1.operational and m1.empty) or
  (m2.operational and m2.empty) or
  (m3.operational and m3.empty) or
  (m4.operational and m4.empty)  prio 1;
  if (m1.operational and m1.empty) then
    @.move(m1);
  elseif (m2.operational and m2.empty) then
    @.move(m2);
  elseif (m3.operational and m3.empty) then
    @.move(m3);
  elseif (m4.operational and m4.empty) then
    @.move(m4);
  end;
end;
```

Run this simulation for 50 days. The output is about 72,000 parts. The Sankey diagram now reflects the availability of the machines (the lower the availability, the fewer parts run across the machines).

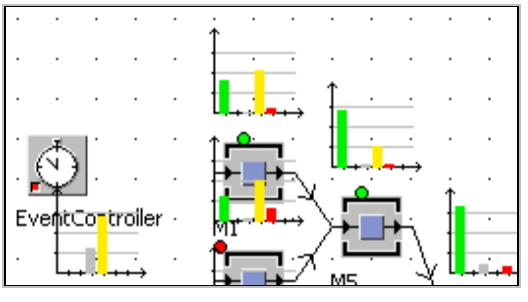


9.3.3 The Bottleneck Analyzer

The Bottleneck analyzer visualizes the default statistics for all selected objects. It is quite simple to use. First, make sure that sufficient space is available above the top object. Open the object BottleneckAnalyzer and click the tab Configure. Select object types for which you want to display statistics.



Click the button Analyze on the tab Analyze to create the statistics evaluation.



The statistical data is displayed graphically in the frame. You can also output the data after the analysis as a table. Click **RANKING TABLE – OPEN**. Once you have chosen a sorting option, the table is displayed.

Sorted according to working time

root.M7

	object 1	real 2	real 3	real 4
string	resource	working	setup	wait
1	root.M7	83.33	0.00	
2	root.M5	70.81	0.00	
3	root.M1	39.62	0.00	

9.3.4 The Display

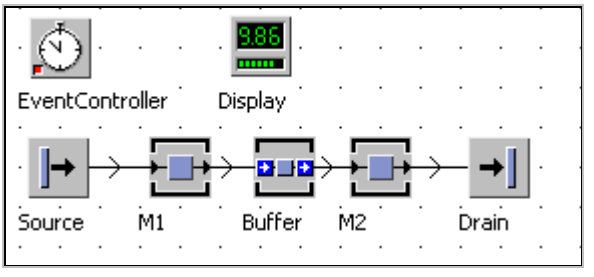
9.3.4.1 Behavior

You can use the display object to show dynamic values (attributes, variables) during a simulation run. The values can be represented as a number or bar. The acti-

vated object periodically checks the value and updates the display (Sample mode) or after a corresponding change (Watch mode). As a bar or pie, the display shows numeric values in relation to the specified interval (between min and max).

Example 101: Display

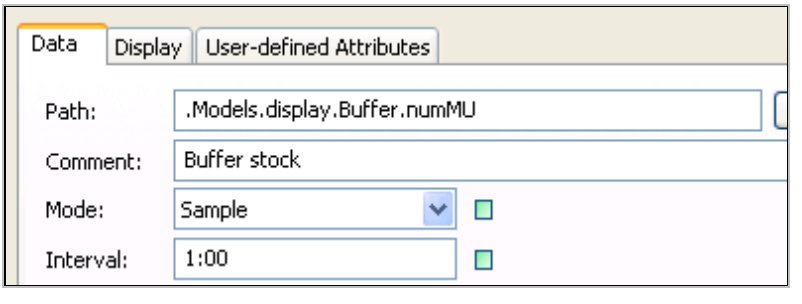
Create the following Frame:



Settings: Source interval: 2:10, M1 processing time: 2:00 availability: 90% 1 hour MTTR, M2 1 minute processing time 50% availability, 2 hours MTTR, Buffer capacity 1,000 no processing time. The display should show the stock of the buffer.

9.3.4.2 Attributes of the Display

Tab Data



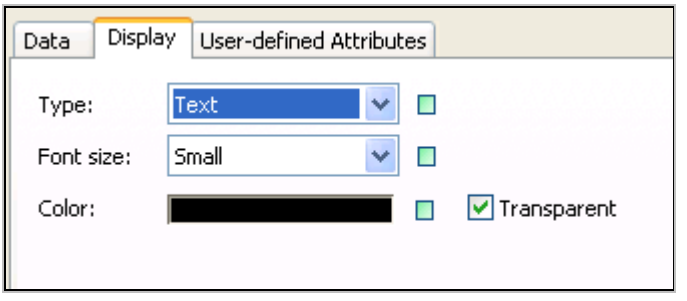
Path: Enter the path to the observed value (relative or absolute). You can enter global variables, attributes, and methods (invalid paths are marked).

Comment: Enter a detailed description of the Display that is displayed under the object.

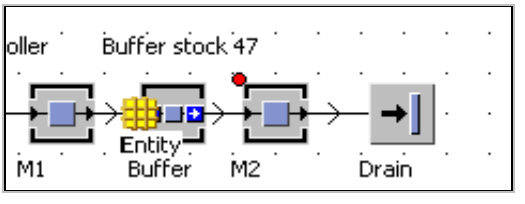
Mode: Select Watch or Sample mode (with interval).

Tab Display

The value of the display can be displayed as a bar (numeric values) or pie, or as text.



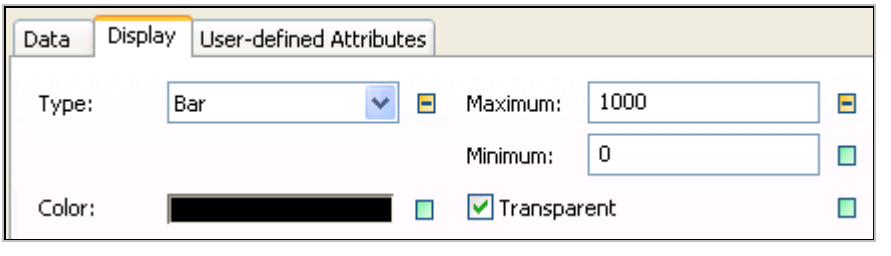
Display as text:



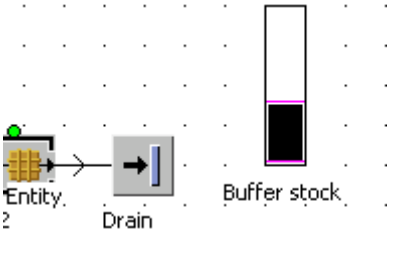
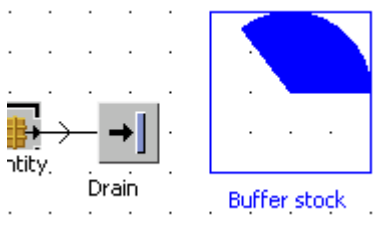
You can only adjust the color and font size and set a transparent background for the display.

Display as a bar/pie

The bar/pie shows the ratio of the actual value and a given maximum value.



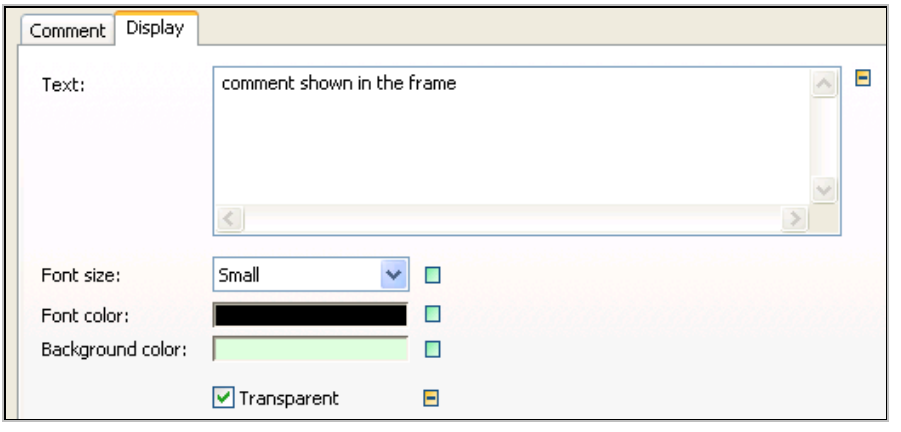
Display:

Bar	Pie
	

9.3.5 The Comment

The comment has no active behavior during the simulation run and can be used for explanations and labeling.

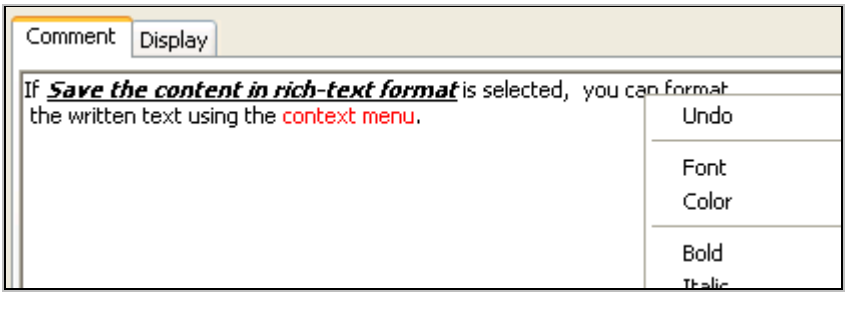
Tab Display



Text: The text, which you enter here, will be shown in the frame. You can assign the text dynamically using the method <path>.text:= <string>.

Font size, font color, background color: Select formatting options for the text of the comment. If Transparent is selected, the comment is shown within a box with the background color shining through.

Tab Comment



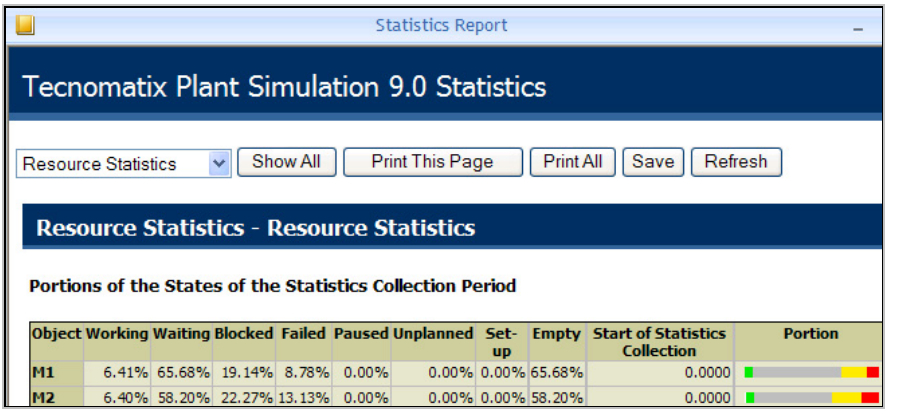
In the big text box on the tab Comment, you can save more text which only is visible after opening the comment object. This text can be created in Rich Text Format (e.g., you create the text in Word and paste it into the comment via the clipboard). You can find formatting options in the context menu of the input field. You can access the contents of the comment with <path>.cont.

9.3.6 The Report

A report can present a very large number of data. The report consists of header data and the report data, which you can arrange hierarchically.

9.3.6.1 Automatic Resource Report (Statistics Report)

You can automatically create reports in Plant Simulation. Select the objects for which you want to create a report by holding down the Shift key. Then press the F6 key.

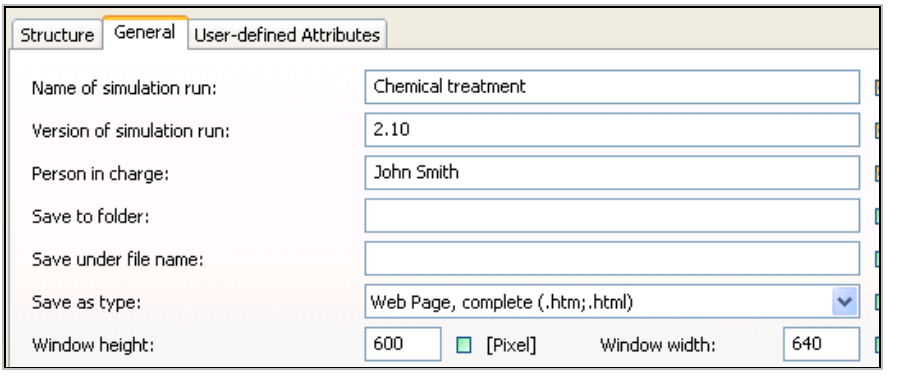


Object	Working	Waiting	Blocked	Failed	Paused	Unplanned	Set-up	Empty	Start of Statistics Collection	Portion
M1	6.41%	65.68%	19.14%	8.78%	0.00%	0.00%	0.00%	65.68%	0.0000	
M2	6.40%	58.20%	22.27%	13.13%	0.00%	0.00%	0.00%	58.20%	0.0000	

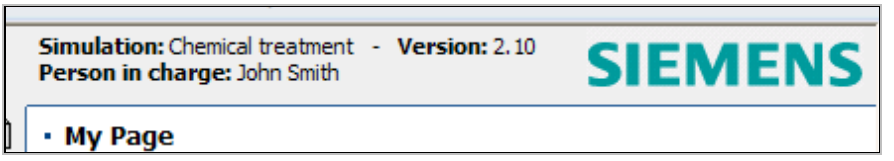
9.3.6.2 Report Header

Example 102: Report

Use the example statistics for creating of the report. Insert a report into the frame, and open the report by double-clicking it. Type in general information about the simulation on the tab General.

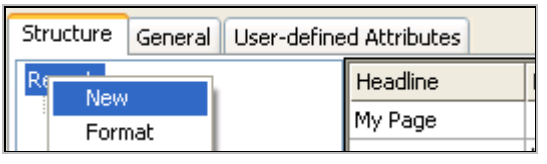


These data are shown later in the report header.



9.3.6.3 Report Data

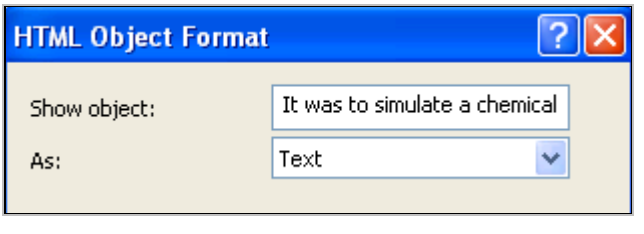
Define the structure of the report and the displayed information on the tab Structure. In the left pane, you can add using the context menu each (fold) object new pages. You must first turn off inheritance. Click the right mouse button on Report and select **NEW** from the context menu:



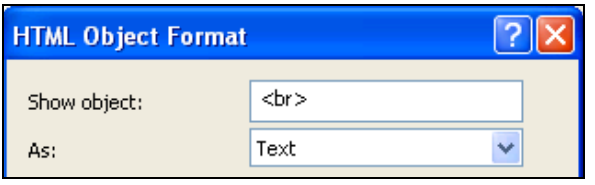
Rename the first sheet of the report to General. Click the entry with the right mouse button. Select **RENAME** from the context menu. Then you can overtype the name of the page. The data of each page will be structured using headings. Each page is separated into three columns. Each column can either contain an icon, text, or an object call.

Example: The first page is to include a brief description of the simulation and a screenshot of the frame. Type "General" in the box Headline, then double-click in the box next to it (column 1). You can place text, icons, or method calls in the report. Select the format text for the first field.

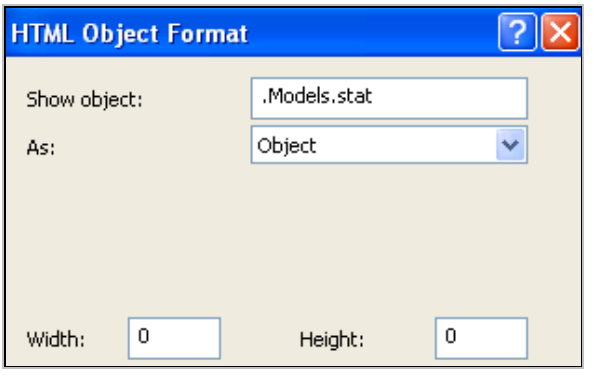
Type the following text into the box Show object: "It was to simulate a chemical treatment, which is supplied by four machines. Through a complex work process, the machines have an availability of only 50%."



*The next line is to remain free. You can insert HTML tags as text; the HTML command for a blank line is
. The report consists of HTML pages. Embedded HTML instructions accordingly modify the appearance of the report. Type the following into the second row, first column:*



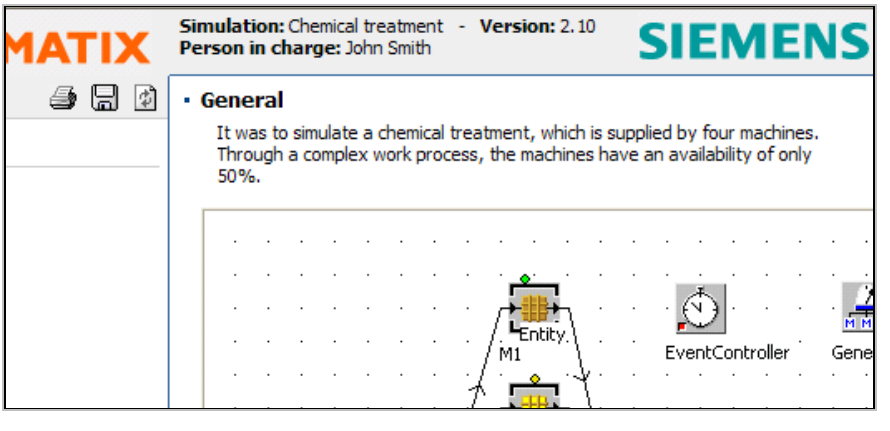
Below is to be shown a screenshot of the frame. Pressing Enter in the last row, last column of the table on the tab structure creates a new row. Enter the following into the first column of the third row:



Enter the address of your frame in the class library. Set width and height both to zero. Plant Simulation then determines the width and height of the image. The structure of the report page should look as follows:

Headline	First column	Second column	Third column
General	It was to simul...		
	.Models.stat		

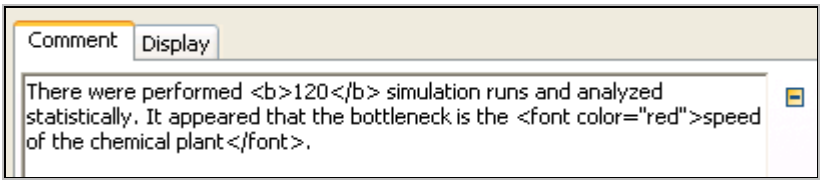
You can view the report by clicking the button Show Report.



9.3.6.4 Texts in Reports

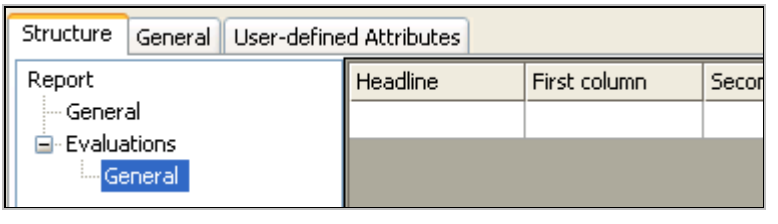
You can enter text directly into the report (see above) or use the comment object for inputting text and output the contents of the comment block in the report.

Example: Insert a comment object into the frame, and type the following into the tab Comment:

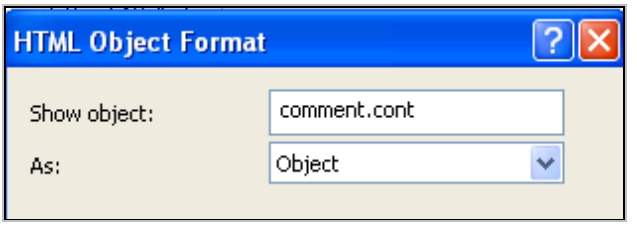


Disable the option in the comment object:

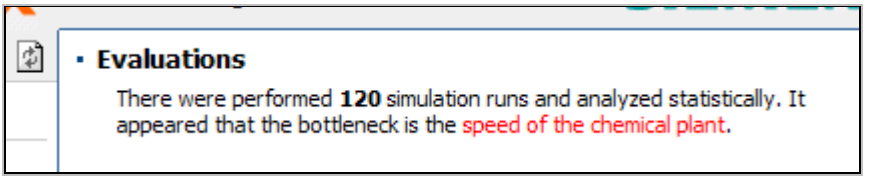
Add a page to the report: evaluations. Then click the right mouse button on the new page and add another page with New (General).



The title of the page should be "Evaluations". Enter the following into the first column of the first row:



The content of the comment is shown with the included HTML formatting in the report.



You can use the following HTML commands for formatting your text (selection).

HTML-tag	*Description*
<H1> Headline 1 </H1>	Headline outline levels 1 to 6
<BIG>Text </BIG>	Rel. enlarged text
<SMALL>Text </SMALL>	Smaller text
<FONT SIZE=7>Text </FONT>	Font size (1 very small to 7 very large)
<FONT COLOR="red">Text </FONT>	Font color as color or hexadecimal
<FONT FACE="Arial, Tahoma"> Text</FONT>	Font face as list separated by commas
<B>Text</B>	Bold
<I>Text</I>	Italic
<U>Text</U>	Underlined text
<S>Text</S>	Strike through
_{Text}	Subscript
^{Text}	Superscript
Combination: <B><I><U> bold, italic, underlined </U></I></B>	

9.3.6.5 Show Objects in Reports

You can access attribute values in reports and display objects. When you display objects in reports, Plant Simulation creates an image of the object (e.g., graph, Frame), or displays values of default attributes. You can type in a complete SimTalk call into the field **SHOW OBJECT**.

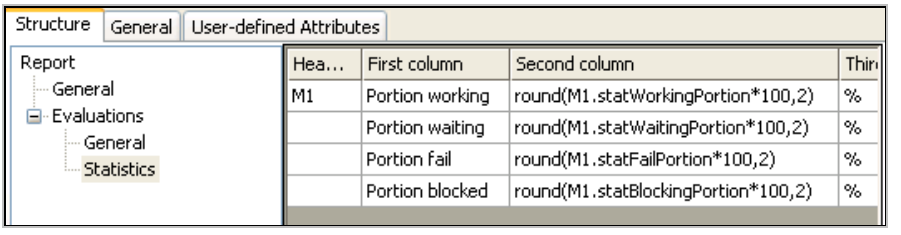
Example: Insert the page "Statistics" into the report. Here you are to display the main statistics of the machines. The headline is the name of the machine. The first column should display the names of the values in the second column and in the third column the units of the values. At the end of the page, a chart with the statistical data is to be displayed. To display the value of statWorkingPortion in the second column, you need to enter the following settings:

*Show object: round(M1.statWorkingPortion*100,2)*

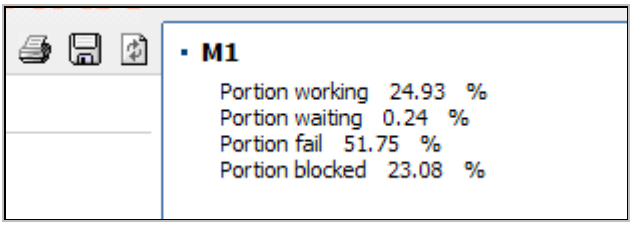
Show as Object

Note:
The default setting in the report is Show as Object. If you want to display text, you must switch to Show Object as Text, otherwise you get an error when calling the report.

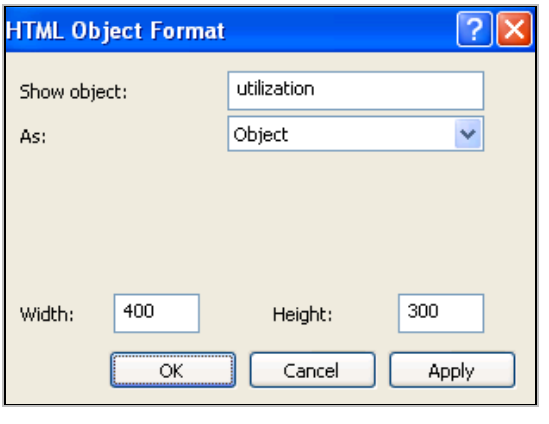
The settings for displaying statistical data for machine M1 look as follows:



Report:



Charts and tables are inserted into the report via a simple object call. When inserting a chart you must specify a size for displaying it. Example: The chart object has the name "utilization", the necessary setting in the report looks as follows:



To display methods, we have to use a little trick. You can access the text of the method as follows:

```
ref(<path>).program
```

The attribute program returns the entire text of the method, including the control characters. The control characters are normally ignored in the HTML display, so there will be a presentation without line breaks and tabs. With the HTML statement <pre>text</pre>, you can force the Report to display line breaks and control

characters. If you want to display a method in the report, use the following setting (display as object):

Headline	First column
Methods	"<pre>"+ref(method).program+"</pre>"

9.3.6.6 Show Images in Reports

You can also display images in reports. The images must be created as icons of the report (Context menu – **EDIT ICONS – ICON – NEW – FILE – OPEN** …). You must specify the icon number when inserting it into the report, for example, the image is saved as an icon in the object Report (icon No. 16):