Co-funded by the
Erasmus+ Programme
of the European Union

AfroTrans project

# Work Package 3: Theoretical and practical teaching materials development for transport in Cameroon and the Democratic Republic of Congo

**T3.5 Logistic management**
**Supply management and logistics - Tutorial**

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them

# MODELING OF LOGISTICS PROCESSES

## ACTIVITY #3

# Defining object parameters

**List of authors:**

**Gdansk University of Technology**

Daniel Kaszubowski, Justyna Staszak-Winkler, Wojciech Kustra

## Contents:

# 1 OBJECTIVE AND NEW SKILLS

A cross-sectional model will be developed using the previously discussed issues related to creating the logic of flows of elements. It will be supplemented with the skills of defining detailed parameters of individual objects and analyzing simulation results. This will provide a better overview of the possibilities of modeling logistics processes.

| **New skills** |
|---|
| Using the *Global Table* for storing input parameters |
| Configuring detailed settings of *Source objects, Queue, Processor, and Combiner* |
| Using the operator group management object – *Dispatcher* |
| Observing statistics from the model level and built-in object statistics |
| Using the *Dashboard* to create charts and tables for analyzing model parameters |

# 2 MODEL ASSUMPTIONS

Fig. 1 shows the target layout of the developed model.

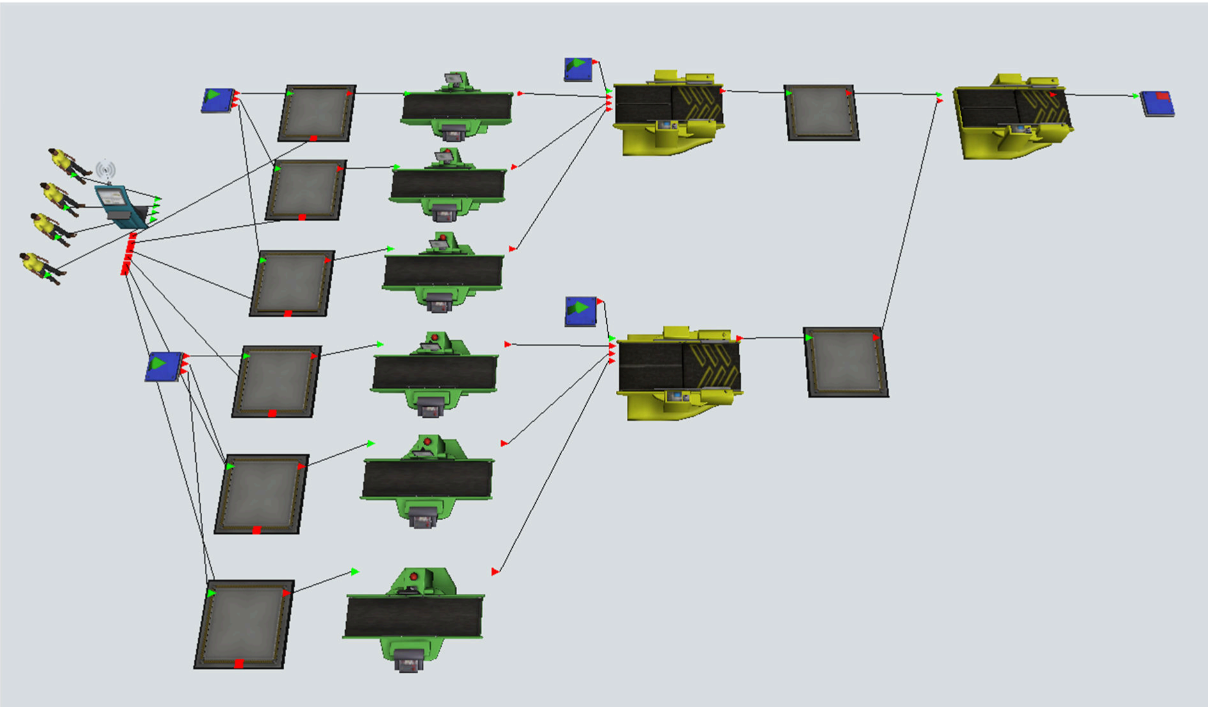


Fig. 1. Target model

1. Two sources generate three types of products
   - *Source1* generates products 1, 2, 3,
   - *Source2* generates products 4, 5, 6.
2. The time of appearance of products on sources follows an exponential distribution with a scale parameter value of 5.

3. Each product is stored on a dedicated buffer before the appropriate *Processor processing station*. The capacity of each buffer is 25 pieces.
4. The processing times on the machines follow an exponential distribution with a scale parameter value of 12 seconds.
5. After finishing processing, the products are sent to buffers and then to the *Combiner packaging machines.*

    - The first machine is used to pack products 1, 2, and 3 – one piece of each of these products goes on each pallet,
    - products 4, 5, and 6 are then assembled into a whole, and the final item is packed onto a pallet.

6. The final item created by combining items 4, 5, and 6 is added to the palette that contains items 1, 2, and 3.
7. Packaging time is 15 seconds, component connection time is 20 seconds.
8. Four operators are working on the production floor, responsible for feeding the machines with products. The time it takes an operator to pick one piece is 0.5 seconds, and the time it takes to put the product back on the machine is 0.75 seconds.

# 3 DEFINING OBJECT PARAMETERS

## 3.1 Creating a data table

All parameters listed in the assumptions will be collected in a single table for easy reference. The Global Table will be used for this purpose*. It is* added to the model in the *Toolbox tab* (next to *Library) by selecting the Global Tables section, then clicking the* green cross. The number of columns and rows can be defined in the properties available on the right side of the screen (Fig. 2).

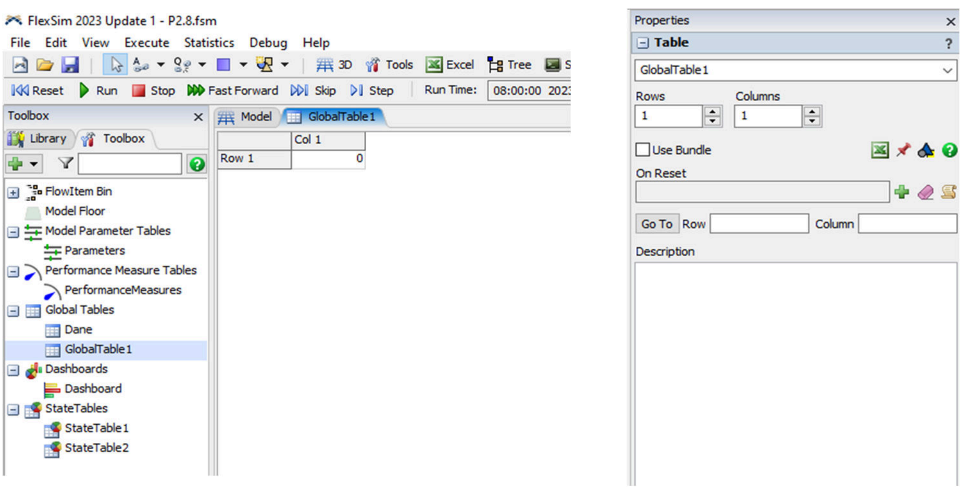



Fig. 2. Initial layout of the new *Global Table*

The name of the created table should be changed to *Data,* and the number of rows should be set to 6, with the names and values as shown in Fig. 3. Objects used later in the model will retrieve data directly from the table rather than entering it manually.

| | Col 1 |
|---|---|
| CzasMiedzyProduktami | 5 |
| CzasPrzetwarzania | 12 |
| CzasPakowania | 15 |
| CzasLaczenia | 20 |
| CzasZaladunku | 0.50 |
| Wyladunku | 0.75 |

Fig. 3 Table *Data*

## 3.2 Flow element

The standard flow unit – *Box will be used in the model* (Fig. 4). In the *FlowItemBin section, the* ProductType label should be added to it.

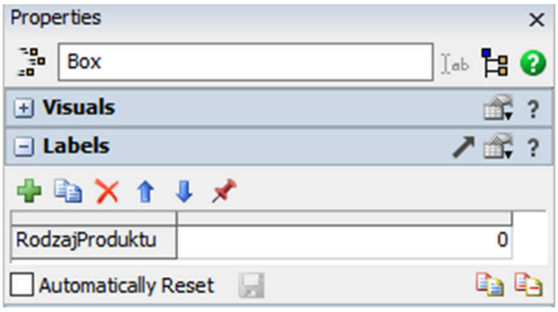



Fig. 4 *Box* - data label

## *3.3* Source – *Source*

*Box* flow units. After configuring the first one, you can copy it. The following source parameters are set in its *Source option* :

1. ***FlowItem Class*** - indicates the flow element being used. *The box should be selected.*
2. ***Arrival Style –*** defining the method of generating flow elements. The default is *Inter-Arrival Style, i.e.,* specifying the time interval between successive elements. This option will be used in the model (Fig. 5). It is also possible to generate elements according to the adopted schedule *(Arrival Schedule)* or in sequence *(Arrival Sequence)*.

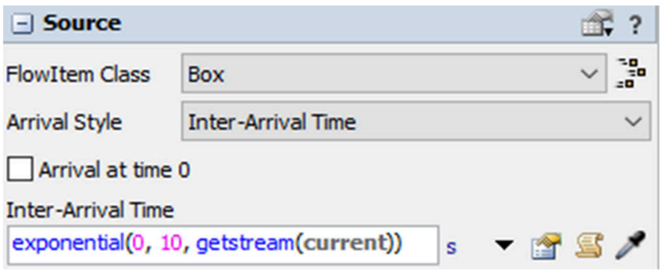


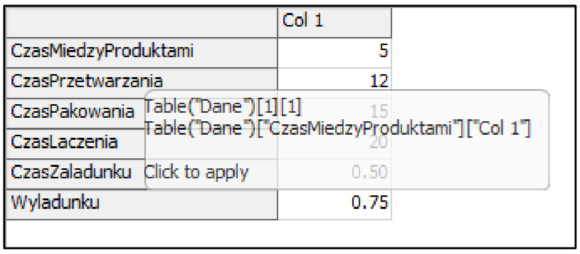

Fig. 5 *Source* – generating flow elements

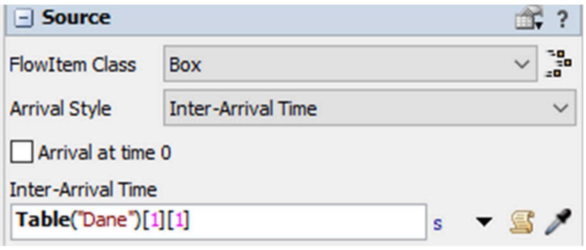
The default notation after selecting the *Inter-Arrival Time option is exponential (0, 10, getstream(current)* ) . This means a random number generated according to the exponential distribution with a scale parameter value of 10. The time between elements will be 10 seconds on average. Additionally, selecting the *Arrival option at time 0* will create a flow element when the model runs. The time between the arrival of subsequent elements can also be specified directly in seconds by entering a value in *Inter- Arrival Time*.

Entering values directly into objects can be troublesome when there are more of them. This can be solved by retrieving the value of a given parameter from a previously created global table. The sampler symbol next to the Inter-Arrival *Time window serves this purpose.*

The picker should indicate the table with data, then the appropriate cell, and confirm (Fig. 6). In the Inter-Arrival *Time field, the entry* `Table ("Data")[1][1]` will appear. This means that data has been collected from the cell at the intersection of the first row and the first column of the table named *Data* Fig. 7. While working on the model, you can modify the parameters of many objects simultaneously by changing only the input data in the table.

Fig. 6 Download data from the table. Fig. 7 Data download result




Retrieving data from a table can also be used to modify the statistical distributions that generate model elements. At *Inter- Arrival Time*, expand the available options and select *Statistical Distribution→Exponential*. The window in Fig. 8 allows you to modify the parameters of the selected distribution, in this case, exponential.

Next to each parameter field, there is a sampler symbol. Using it, you can retrieve the value of the *Scale* parameter of the distribution from the *Data table*. The result is shown in Fig. 9.

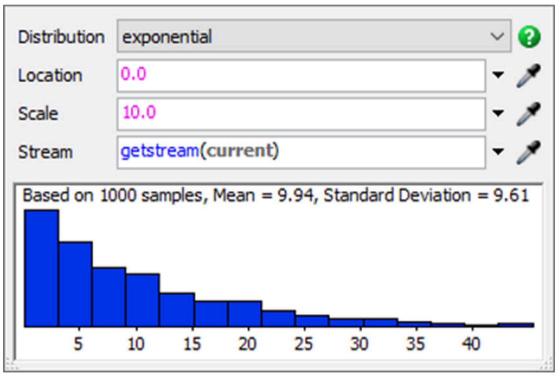



Fig. 8 Exponential Distribution Options

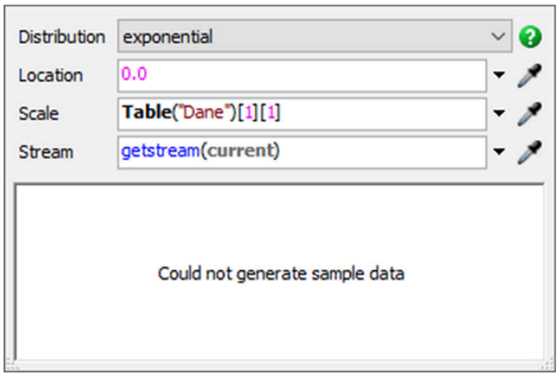



Fig. 9. Modified exponential distribution

### 3.4 Buffer - *Queue*

The buffer (or queue) is used to store flow elements. It has several settings that allow you to modify its behavior (Fig. 10):

1. ***Max Content*** – specifies the buffer capacity in pieces.

2. ***Item Placement*** – the way the flow elements are arranged. By default, *the Stack Inside Queue option is selected*, which means that they are stored in successive layers. *The Stack option, Vertically, means stacking* one element on top of another. *Horizontal Line, on the other hand,* allows you to stack elements one behind the other. *Do Nothing*, on the other hand, places all elements in the same location, without visually distributing them within the object.
3. ***Stack Base Z*** – specifies the level in the XYZ coordinate system from which the elements in the buffer are arranged. An example after setting the height to 0.5 m is shown in Figure 11.

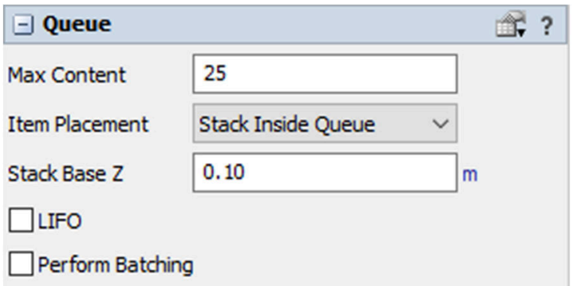


Fig. 10 Basic options of the Queue buffer

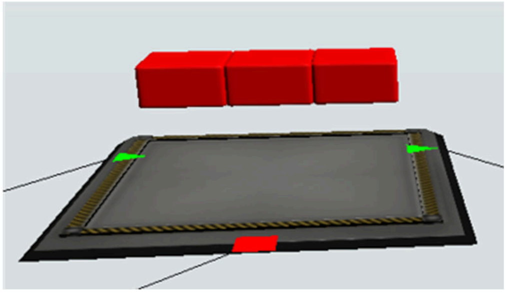


Fig. 11. Incorrect StackBase Z setting

4. ***LIFO –*** specifies the rules for assigning priority to items in the queue. By default, they are sent according to ***the FIFO principle*** *(First In,* First Out*).* The elements are then handled in the order they are entered into the buffer. ***LIFO selection*** *(Last in, First out)* will cause the items that entered the queue last to be served first.
5. ***Perform Batching*** – allows you to create a batch whose size is determined by *the Target Batch Size*. If the buffer contains the specified number of elements, they will be sent to the next object. In *the Max Wait Time field, you can specify the maximum time to wait before building* the batch. Entering "0" means the elements will be sent to the specified object only after the batch is complete. On the other hand, selecting *the Flush contents option between batches* means the buffer can contain only elements from the same batch (Fig. 12).

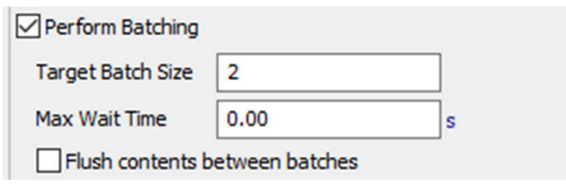


Fig. 12 *Perform option Barching*

### *3.5* Service Station - *Processor*

The service station is used to represent various technological processes on flow elements. This mainly involves delaying the sending of the element to the next object in the model, and is also related to the possibility of connecting the operator via the central port. The object has the following options:

1. ***Max Content*** – defines the capacity of the object; "1" is most often used in practice.
2. ***Animate Items*** - an animation of the workpiece moving along the machine belt is started. Otherwise, the element is stationary at its central point.
3. ***Setup Time*** – machine changeover time. Can be entered directly or taken from the table. Additionally, you can indicate an operator necessary for changeover, independent of the operator performing the basic processing operation.
4. ***Process Time*** - processing time is defined in the same way as set-up time.

In the considered model, time will be generated using an exponential distribution, the scale parameter value of which will be taken from the *Data table* (Fig. 13)

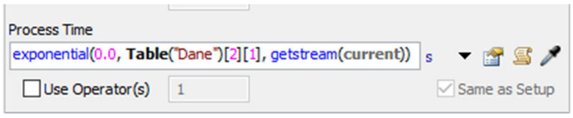



Fig. 13. Setting the processing time

### 3.6 Combiner

The adder is used to pack or combine flow elements. The object is derived from the service station, so it has a *Processor tab* with options identical to the standard machine. The difference is the inability to define capacity, which is always equal to 1.

Two adders should be added to the model, and their processing times should be defined. This will consist of retrieving the appropriate value from the *Data table* (Fig. 14).

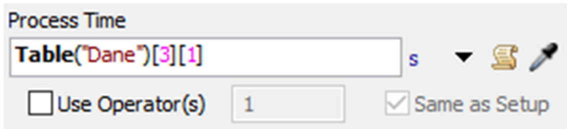



Fig. 14 Processing time settings

In the Combiner tab, you can select its operating mode. In each case, the device takes exactly one element sent via input port 1. If the elements are to be placed on a pallet, they should therefore be connected to this port. Available operating modes are:

1. ***Pack*** – the element sent via input port 1 is packed with the elements sent via the remaining ports according to the specifications given in *the Component List*.
2. ***Join*** – the element sent through input port 1 is joined with elements from the other ports. The element that leaves the adder is the same element that arrived at input port 1.
3. ***Batch*** – a batch is built (similarly to *Queue* ), consisting of exactly one item from port 1.

*The Component* List is available after adding directional connections to objects preceding the combiner – the list of input ports will then be known (Fig. 15).

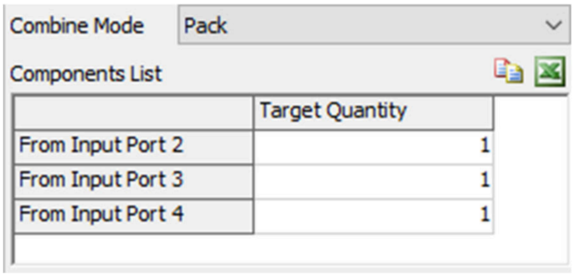



Fig. 15. Component list of input ports

For each adder, a source should be added to the model, with the pallet as a flow element. Assuming the pallet is available at all times, it is sufficient to set the Inter-Arrival *Time* to "0". The operating mode should be set to *Join*. After connecting the three preceding machines, the adder settings are shown in Fig. 16. In the individual rows of the component list, the appropriate values should be entered.

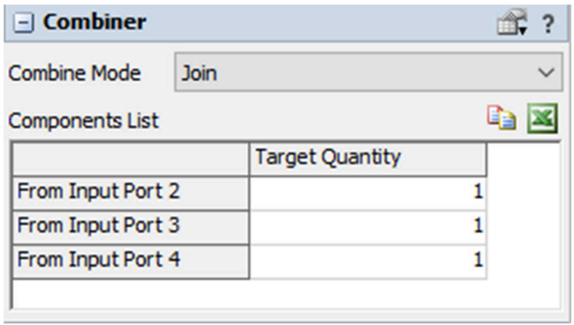



Fig. 16 Component List in the model

# 4 CREATING FLOW LOGIC

## 4.1 Directional connections

Missing connections must be added to the existing, configured model elements, and the logic of the flow of elements must be defined. The connections are shown in Fig. 17.

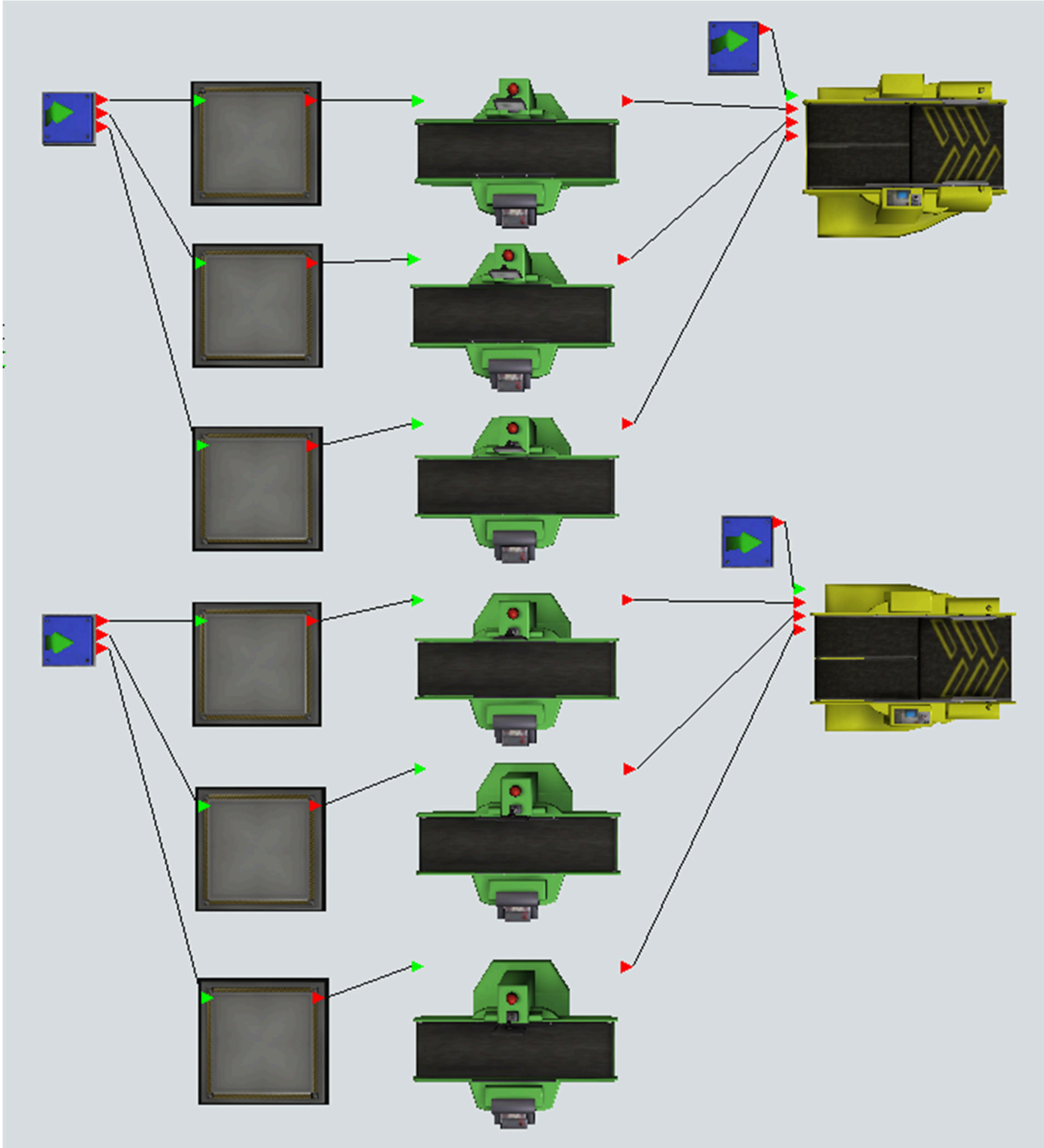


Fig. 17 Directional connections in the main part of the model

The first step in creating the flow logic will be to define both sources. For the first source in *the Triggers section,* we choose *On Creation → Data → Set Label and Color.* For the previously added ProductType label, its value will now be defined. The default notation `duniform(1,3,getstream(current))` will be preserved, which means a value generated from a discrete uniform distribution on the interval from 1 to 3. The flow to the next buffers will also be determined based on this label. In the *Send To Port option*, you should specify the item. ProductType ( Fig. 18).

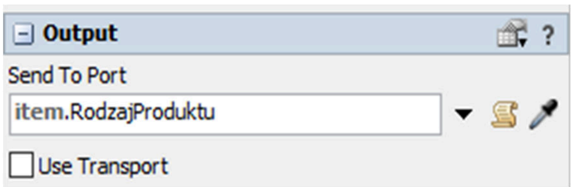


Fig. 18 *Source1* – Output Settings

The second source will generate only products 4, 5, and 6. So it is enough to change the value of the ProductType label (Fig. 19).

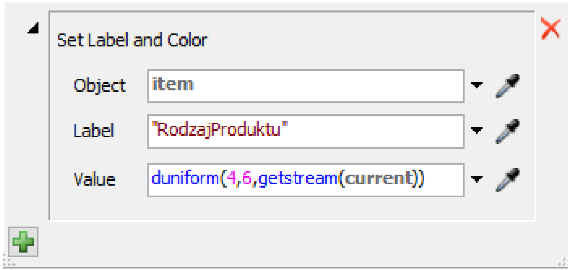



Fig. 19 *Source2* – ProductType label value

The flow of elements from *Source2* will be determined using *the Port By Case function* (Fig. 20).

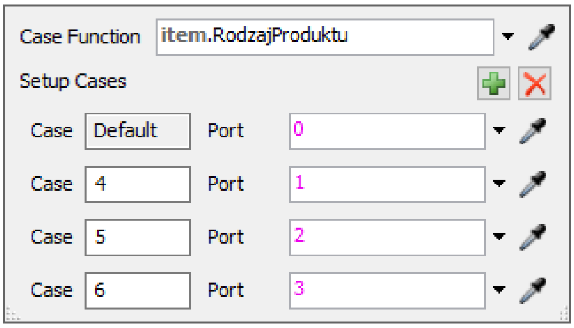



Fig. 20 *Source2* – flow of elements

### *4.2* Transport of elements – *Dispatcher*

Loading products onto machines is performed by a group of operators. If we want to use not a single operator assigned to a specific machine, but a group of operators who operate several machines alternately, we need to add an object that manages operators to the model.
In the library, *Task Executors* contains a *Dispatcher object* that should be added to the model. This is a dispatcher that assigns tasks to subordinate resources. It should be connected via central ports ("S") to buffers located in front of service stations, and remember to mark "Use Transport" *on the queues.* From now on, the dispatcher will be responsible for controlling the transport of material from buffers to service stations using the operators assigned to them.
Four operators should be added to the model, and their dispatcher should be connected to the operators (Fig. 21). For each operator, the time for picking up (*Load Time)* and putting away goods *(Unload Time) should be specified*. As before, these values are taken from the *Data table* (Fig. 22).

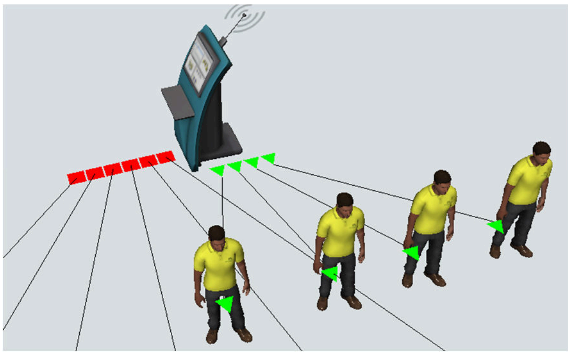


Fig. 21 *Dispatcher connection* with operators

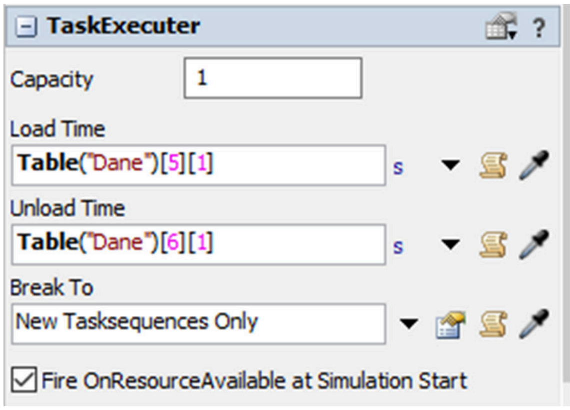



Fig. 22. Service times for operators

### 4.3 Other elements of the model

To finish, a few missing objects need to be added. These are two buffers after the combiners, another *Combiner3,* and the *Sink* element (fig. 22).

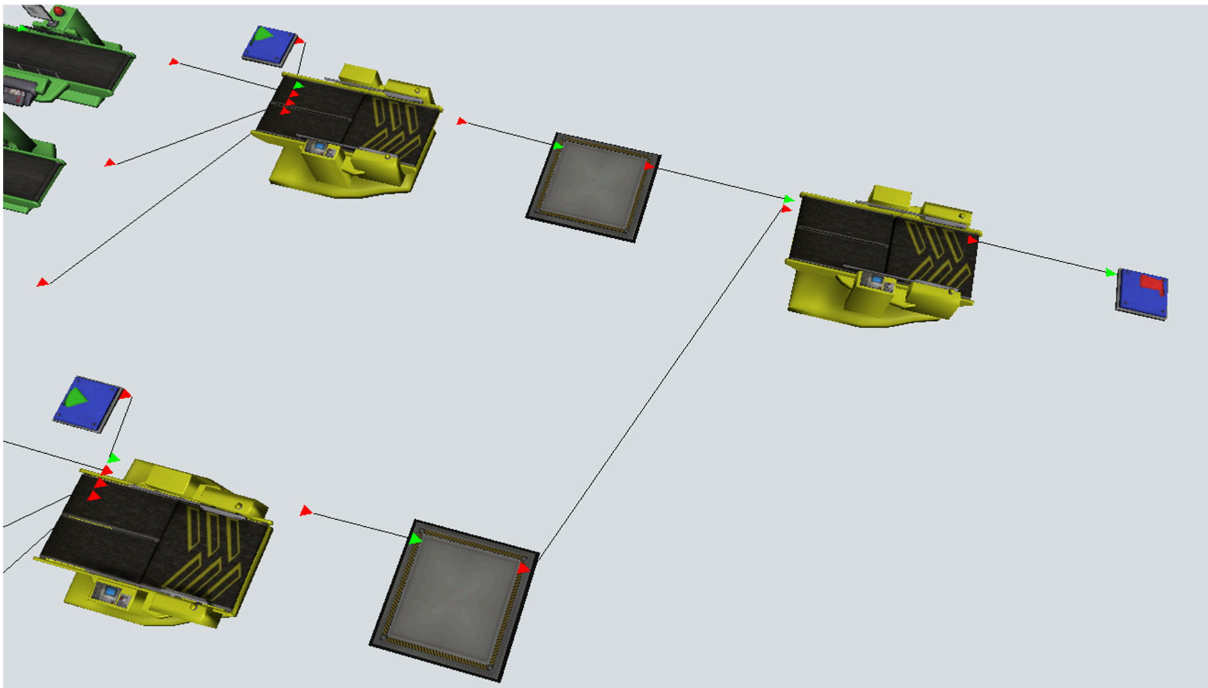


Fig. 22. Other elements of the model

*Combiner3* is a packaging machine. Its input port 1 should be connected to a queue of pallets, each with three pieces of different flow elements. Input port 2 should be connected to a buffer to store brown packages, the result of combining elements 4, 5, and 6.
The next step is to enter the parameters of both buffers and the adder. The buffer capacity is 25 pcs., while the adder works in packing mode with the operation time taken from the table. The settings of both objects are shown in Fig. 23 and Fig. 24.
The simulation result is shown in Fig. 25. *Combiner3* shows one element of types 1, 2, and 3, and one piece (brown) of the element that is a combination of types 4, 5, and 6.

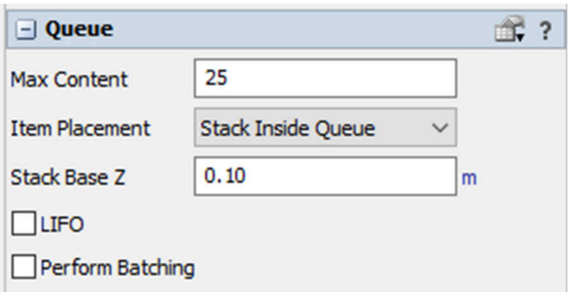



Fig. 23. Buffers before *Combienr3*

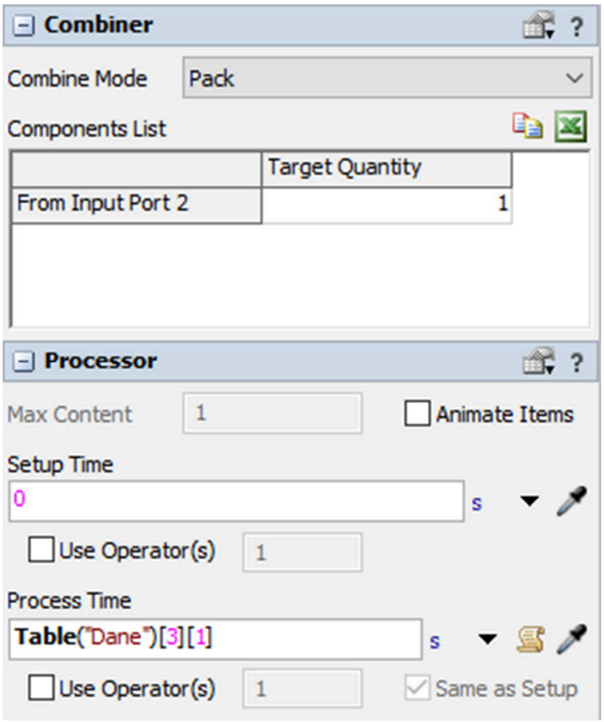



Fig. 24. *Combiner3 settings*

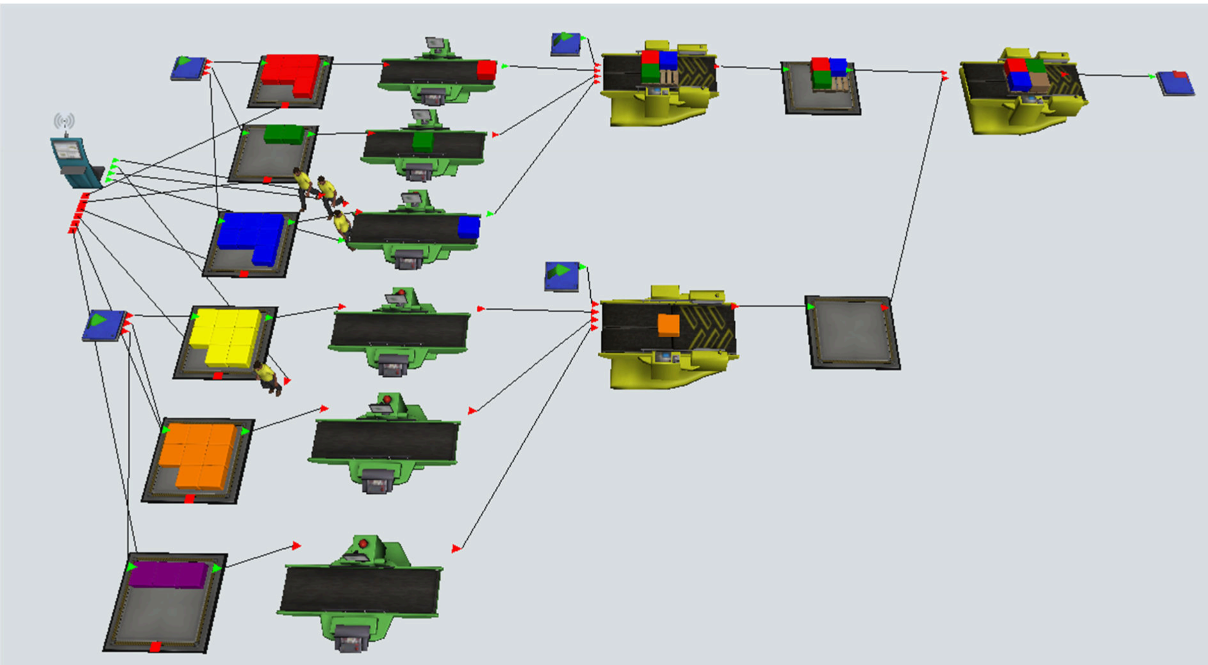


Fig. 25. Simulation result

# 5 ANALYSIS OF SIMULATION RESULTS

## 5.1 Model-level results

The model construction and simulation observations aim to provide information on the quantitative characteristics of the modeled system and its individual elements. The results can be read from the model plane, using built-in object functions or using the *Dashboard control panel*.

It is worth starting the reading of parameters from objects by setting the simulation duration, for example, to 1 hour, i.e., 3600 seconds – options *Run Time → Stop Times* (Fig.26)

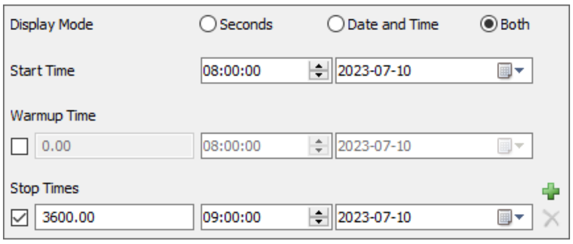



Fig. 26. Simulation duration

The most basic way to observe statistics is to read them from objects. To do this, go to *Properties* in the *View section, Styles,* select *Show Names and Stats*. The effect of this setting is shown in Figure 27.

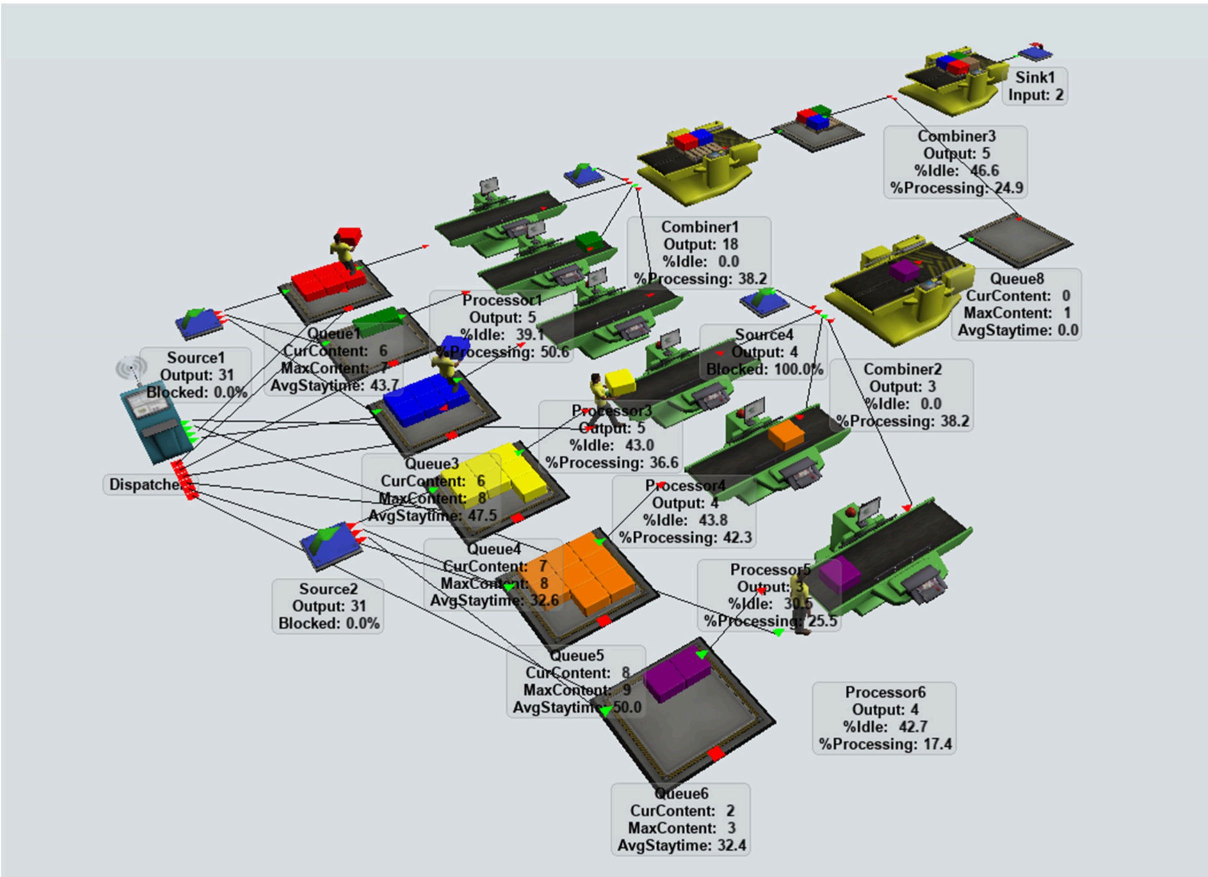



Fig. 26. Displaying statistics directly on model objects

## 5.2 Data collection by objects

Each object collects internal statistics that can be viewed in the Statistics section. They include (Fig. 27):

1. ***State*** – the current state of the object.
2. ***Throughput*** – flow: the number of elements that entered the object *(input)* and were sent further *(output)*.
3. ***Content*** – content: the number of elements that are on the object – at a given moment *(Curr),* the smallest *(Min),* the largest *(Max),* and the average *(Avg)* number of elements that were on the object at the time of observation.
4. ***Staytime*** – the time the element spent on the object. You can read the minimum *(Min)*, maximum *(Max),* and average *(Avg )* value of this parameter.

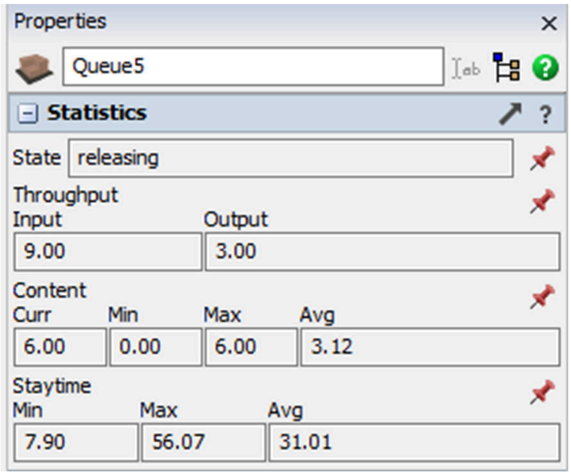



Fig. 27. Sample object statistics

 Each statistic has a pin icon on the right side. Selecting it will give you the option to add the statistic to the chart on the *Dashboard panel* (Fig. 28). The result will be a line chart (Fig. 29) and a new Dashboard window *with a Library assigned to it* for formatting charts.

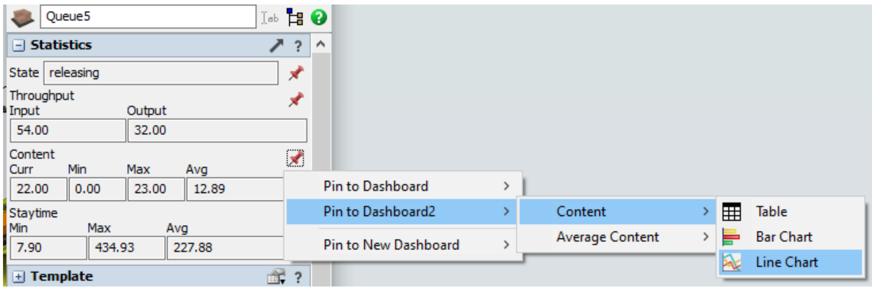



Fig. 28. Adding data to the chart

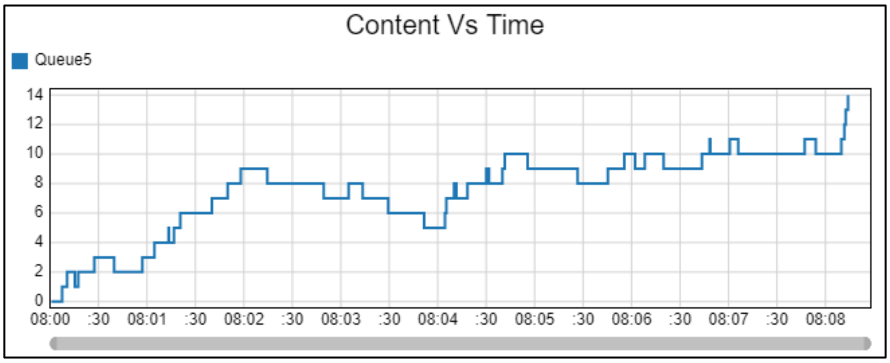



Fig. 29. Line chart with object parameters

### 5.3 *Dashboard*

In addition to the method presented earlier, you can also add charts and tables to *the Dashboard* in other ways. This involves selecting *Dashboard* from the menu bar and then selecting *Add Blank Dashboard.* A new window will appear, where you can first select the type of chart or table (Fig. 30).
For example, you can select *State* → *State* → *Pie Chart from the group.*

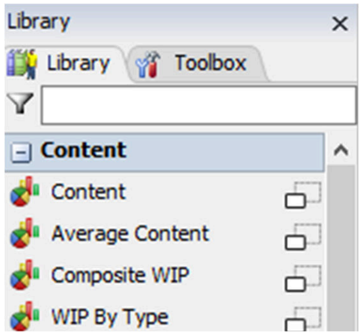
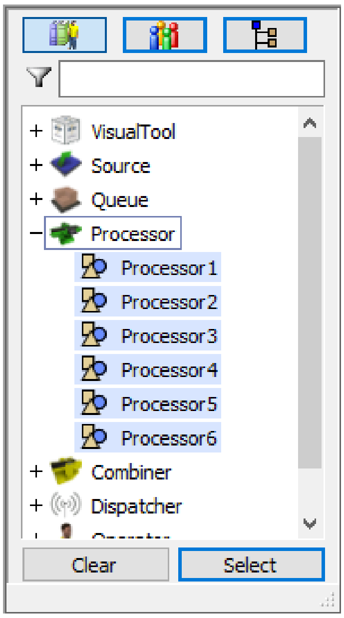



Fig. 30. Fragment of *the Dashboard window*

A chart template has been added that will illustrate the performance of selected resources. Objects to the chart are added with the green button in the Options section of the chart, then select objects (Fig. 31). The result of adding selected objects to the chart is shown in Fig. 32.



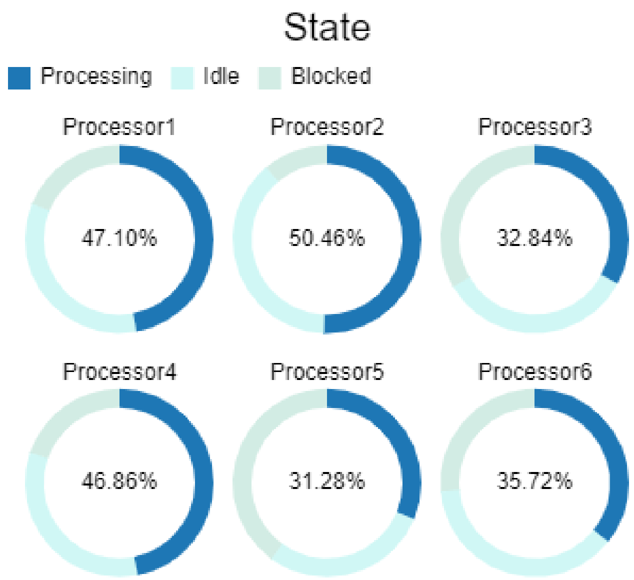




Fig. 31 Selecting objects

Fig. 32 Pie chart for workstation performance