ŠKODA AUTO University

Computer Simulation of Logistics Processes

Plant Simulation Software Structure

Jan Fábry

20/03/2023

Plant Simulation Software Structure

ŠKODA AUTO University

Aim of the lecture

- Introduce the simulation software “Plant Simulation 16” and its basic structure.

Plant Simulation Software Structure

ŠKODA AUTO University

Structure of the lecture

- Directive VDI on the simulation studies.
- Program launching, initial program window.
- Available objects, libraries.
- New model, material flow objects, resources, information flow, user interface, mobile units, tools.
- Models, name syntax.
- Princip of inheritance, key terms, class and instance, class and subclass, identification and view of the inheritance structure.
- View of the object status.
- Movement strategy of MUs, object’s exit strategy.

Plant Simulation Software Structure

ŠKODA AUTO University

Standard VDI on simulation studies

- The process of simulation studies is described by standard VDI* 3633. That splits simulation project into three phases:
  - Preparation.
  - Realization.
  - Evaluation.

- Deciding, whether the simulation is appropriate for the project realization.
- Formulation of study and its goals.
- Estimate of complexity (costs, time).
- Ensuring of necessary data, its preparation and validation.
- Rough analytical estimate.
- Creation of conceptual model and its verification.

Note * - „VDI“ – Verein Deutscher Ingenieure – Association of German engineers and naturists, which methodically covers a significant number of scientific fields and disciplines. It has over 150 000 members.

Plant Simulation Software Structure

ŠKODA AUTO University

Standard VDI on simulation studies

- The process of simulation studies is described by standard VDI* 3633. That splits simulation project into three phases:
  - Preparation.
  - Realization.
  - Evaluation.

- Creation of virtual (simulation) model and its validation.
- Planning of simulation experiments.
- Realization of experiments.

Note * - „VDI“ – Verein Deutscher Ingenieure – Association of German engineers and naturists, which methodically covers a significant number of scientific fields and disciplines. It has over 150 000 members.

Plant Simulation Software Structure

ŠKODA AUTO University

Standard VDI on simulation studies

- The process of simulation studies is described by standard VDI* 3633. That splits simulation project into three phases:
  - Preparation.
  - Realization.
  - Evaluation.

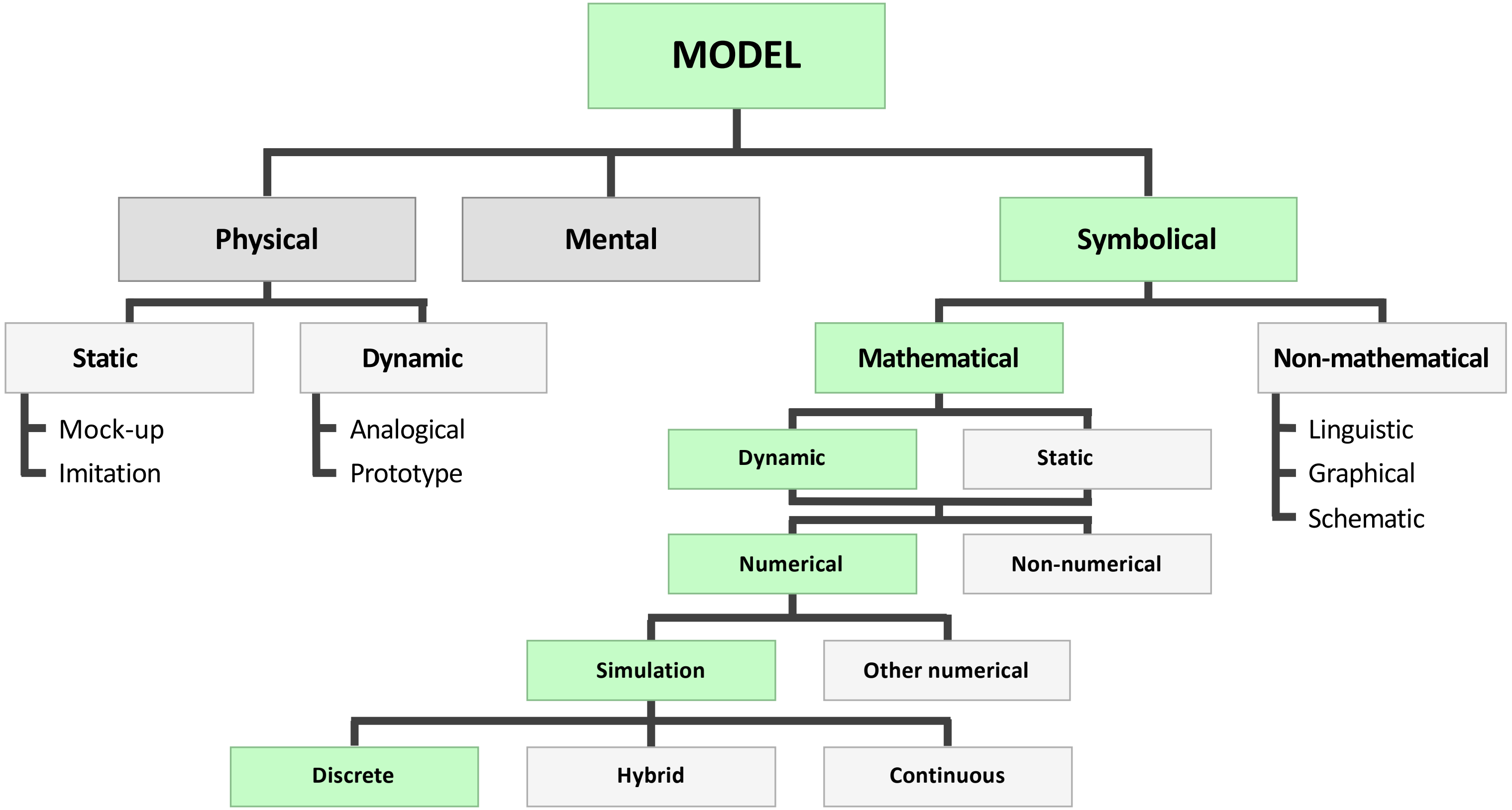
- Processing of results.
- Interpretation of results.
- Documentation.

Note * - „VDI“ – Verein Deutscher Ingenieure – Association of German engineers and naturists, which methodically covers a significant number of scientific fields and disciplines. It has over 150 000 members.

Plant Simulation Software Structure

ŠKODA AUTO University

Classification and integration of the software into models structure

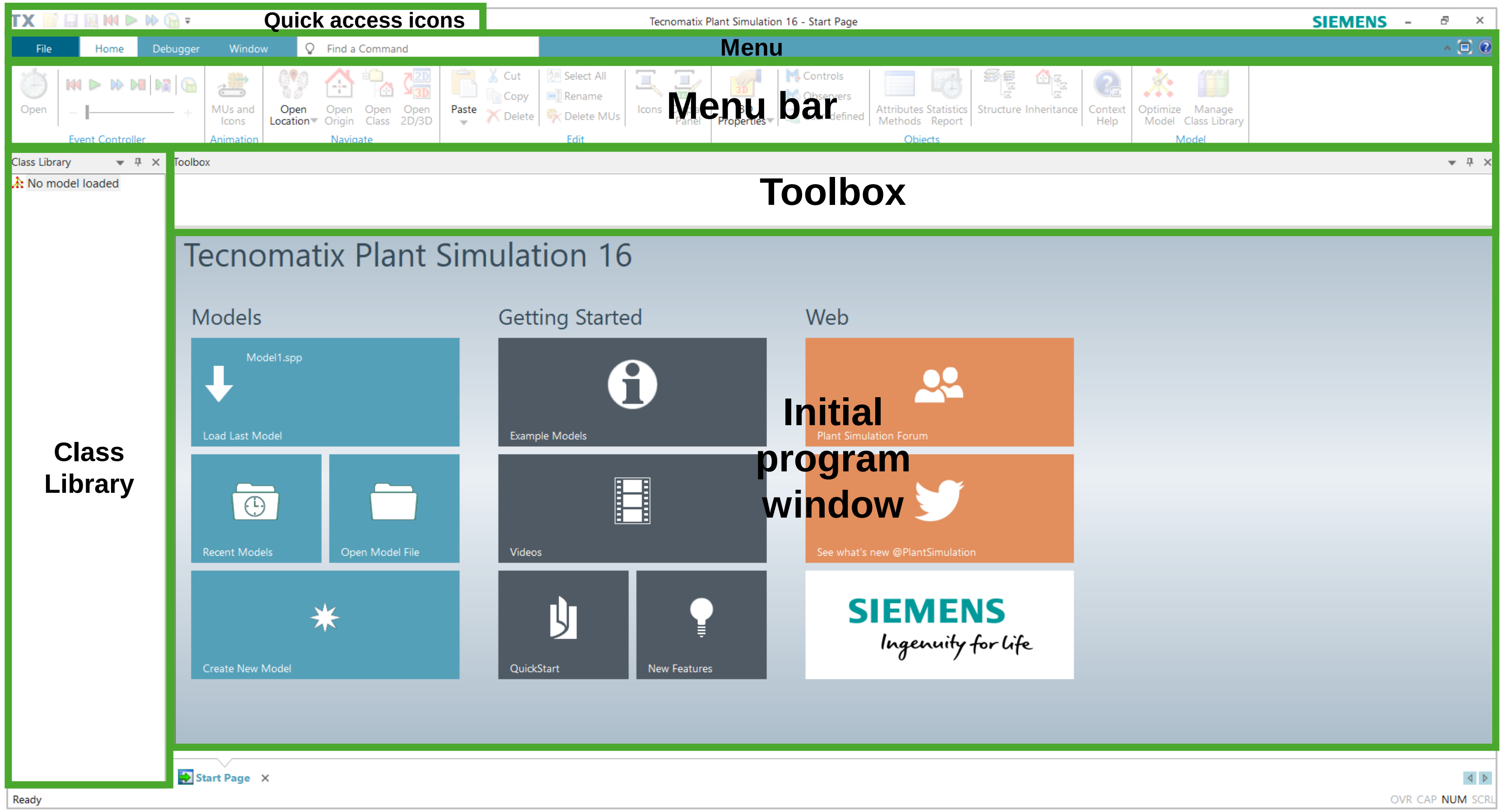


Source: DANĚK, Jan. HUMUSOFT S.R.O. *Využití simulace jako inženýrského nástroje během životního cyklu výrobků a procesů*. Praha, 2007, 70 s.

Plant Simulation Software Structure

Initial program window

- This program window will be open after initializing of software. Its look can be modified in the menu ”Window”.

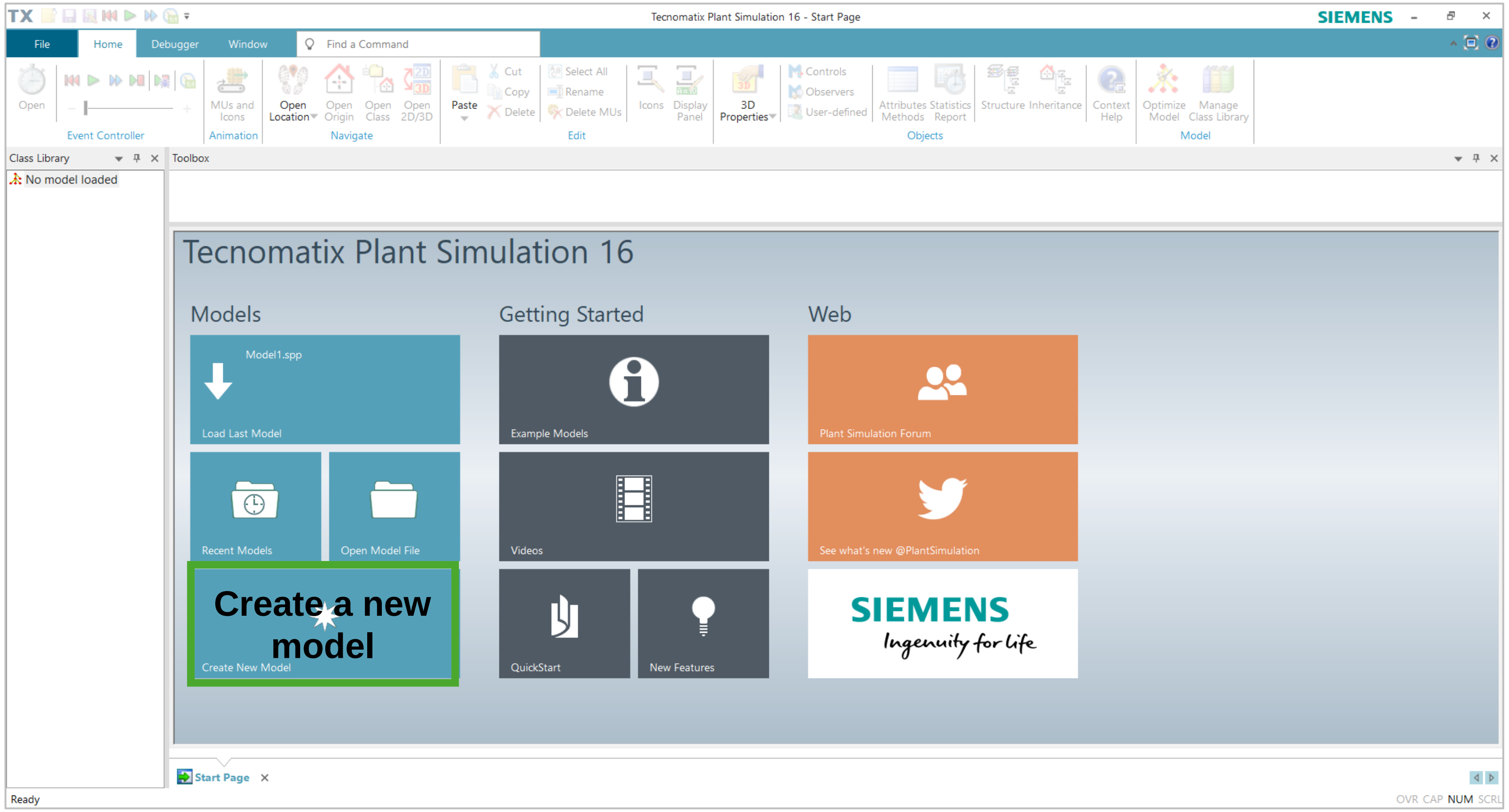


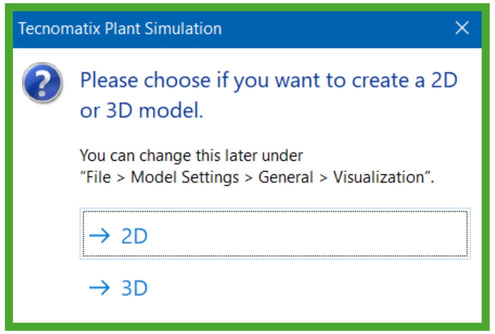
ŠKODA AUTO University

Plant Simulation Software Structure

Initial program window

- Before creating a new model, the user is asked whether they want to create a model in a 2D or 3D environment.



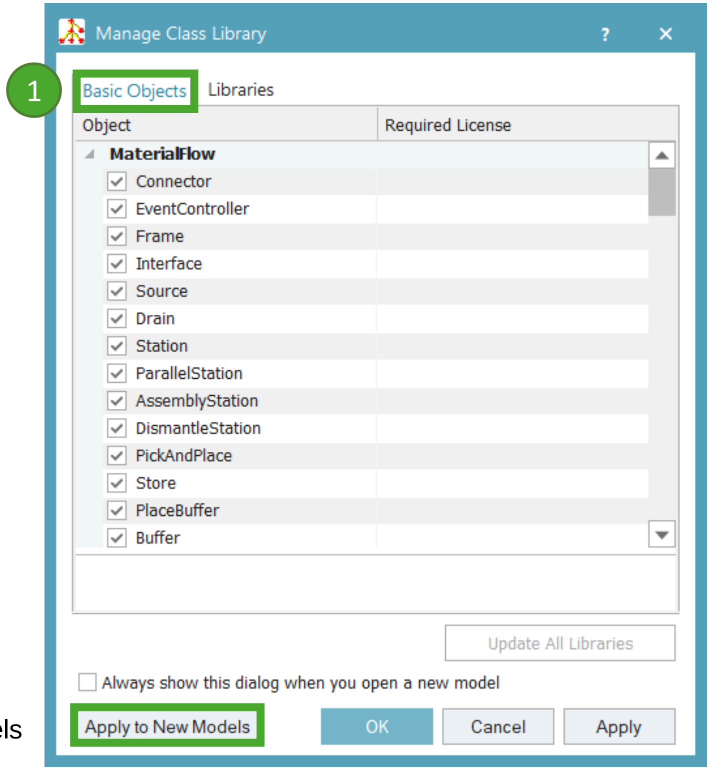


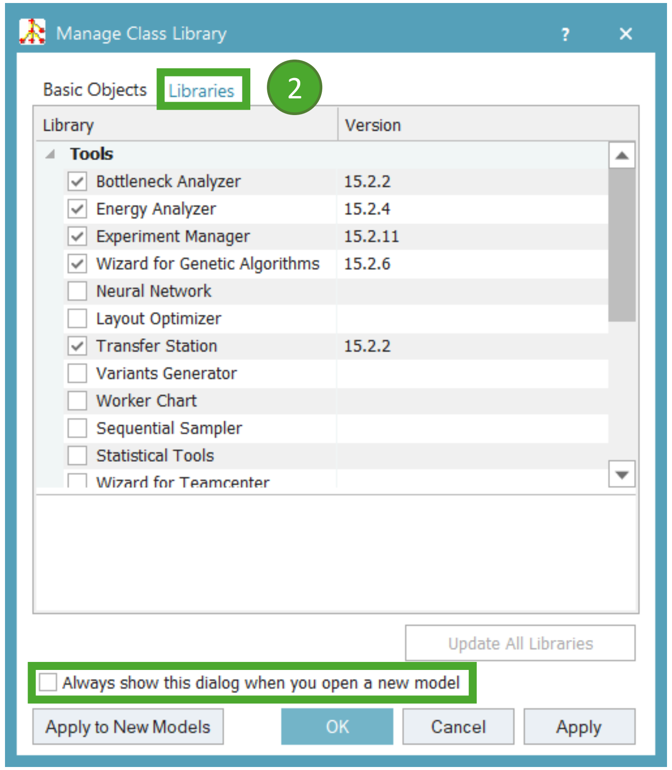
Plant Simulation Software Structure

ŠKODA AUTO University

Available objects, libraries

- User is asked, which objects are wanted to be as a standard in the tab "Basic Objects"(1) before launching of a new model. Tab "Libraries"(2) allows to extend the next libraries of elements, interfaces, graphs, etc.





Use for new models

Show this dialog whenever a new model is created.

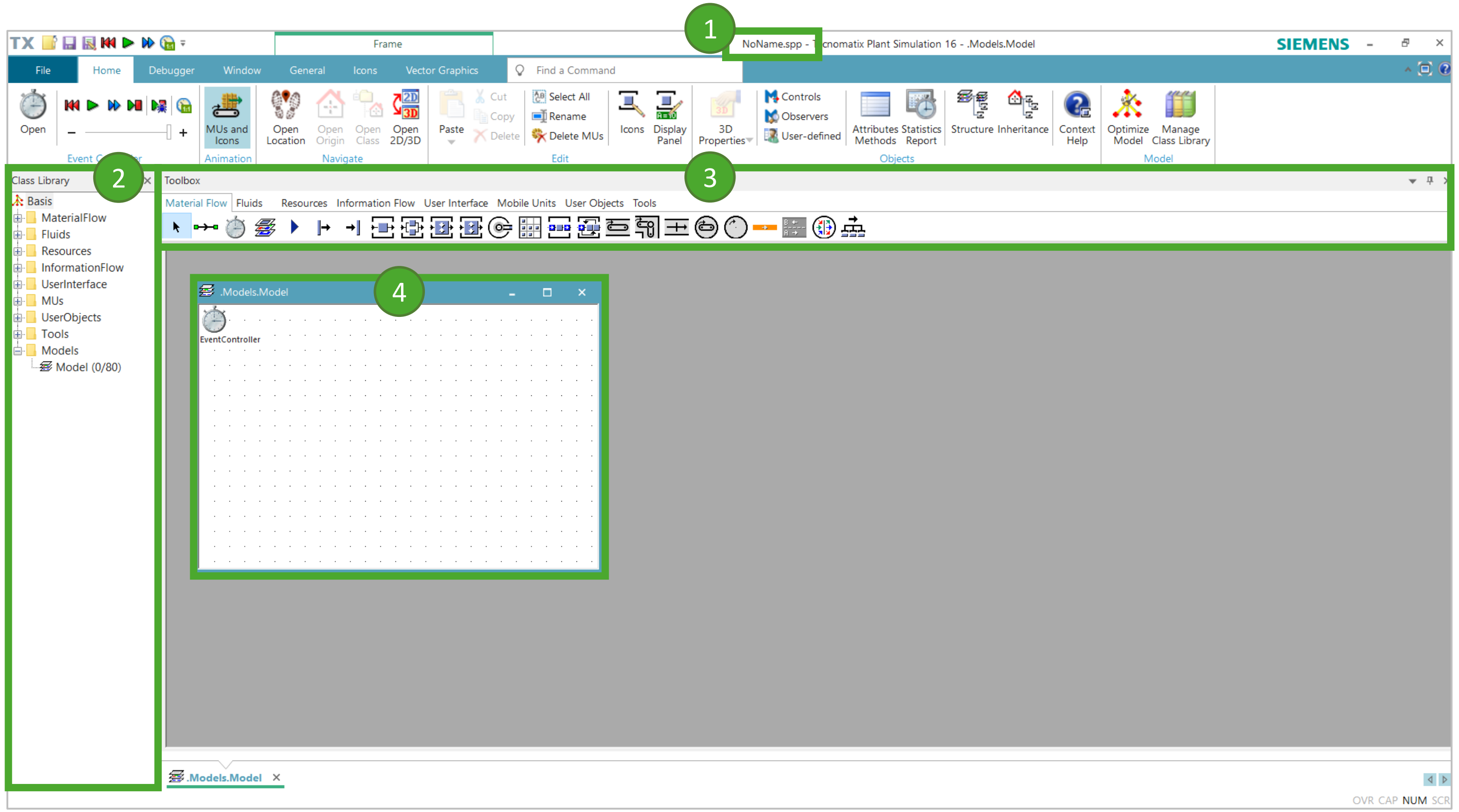
Button "Apply" is in general for confirmation of choices in the Plant Simulation. After that, we can press the button "OK". Here, however, we select the „OK" button directly.

Plant Simulation Software Structure

ŠKODA AUTO University

New model

- Files of the program Plant Simulation have the suffix *.spp (1).
- The file has always Class library (2) based on hierarchy. Individual folders contain class of objects. The class structure can be modified and changed.
- Some class of objects can be found in the Toolbox (3). Here it is possible to create a new folder for saving of models and their parts.
- Each new file has a new "Frame" (4). Here objects or model can be created.



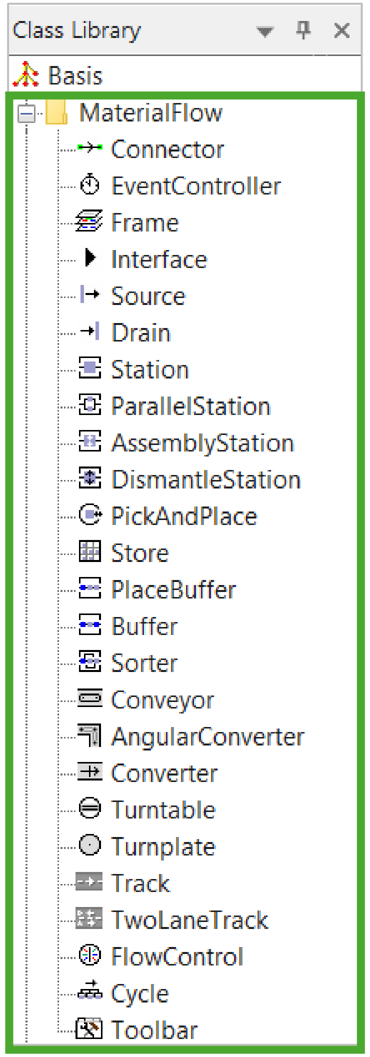
Plant Simulation Software Structure

Material flow objects

- The objects of material flow modelling – “MaterialFlow”.
- We can find ”active objects” among objects of the material flow, which transport, process or handle mobile units (e.g. products in the model).
- On the other hand, ”passive objects” are more likely as the means for transport (communication) or for storing (buffers).

Elements of the material flow:

- Connector
- EventController
- Frame
- Interface
- Source
- Drain
- Station
- ParrallelStation
- AssemblyStation
- DismantleStation
- PickAndPlace
- Store
- PlaceBuffer
- Buffer
- Sorter
- Conveyor
- AngularConverter
- Converter
- Turntable
- Turnplate
- Track
- TwoLaneTrack
- FlowControl
- Cycle



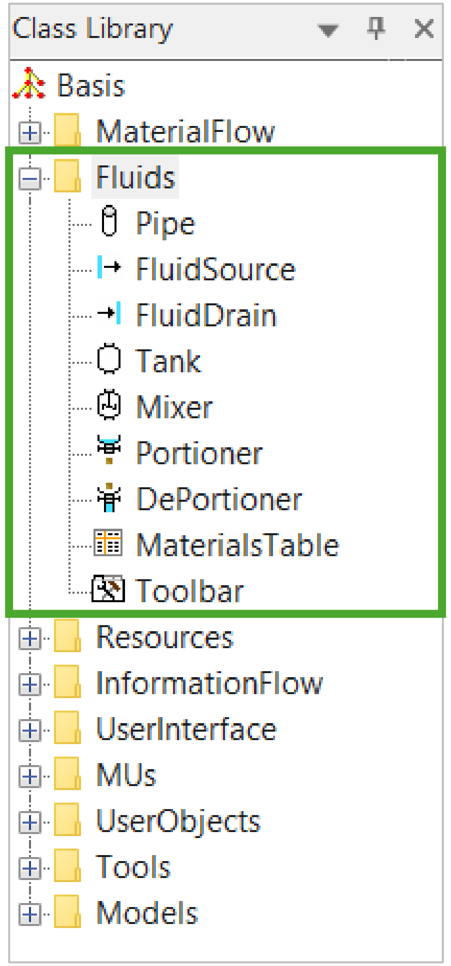
Plant Simulation Software Structure

Fluids

- The “Fluids“ folder is used to model the flow of fluids.

Elements of the fluids:

- Pipe
- FluidSource
- FluidDrain
- Tank
- Mixer
- Portioner
- DePortioner
- MaterialsTable



Plant Simulation Software Structure

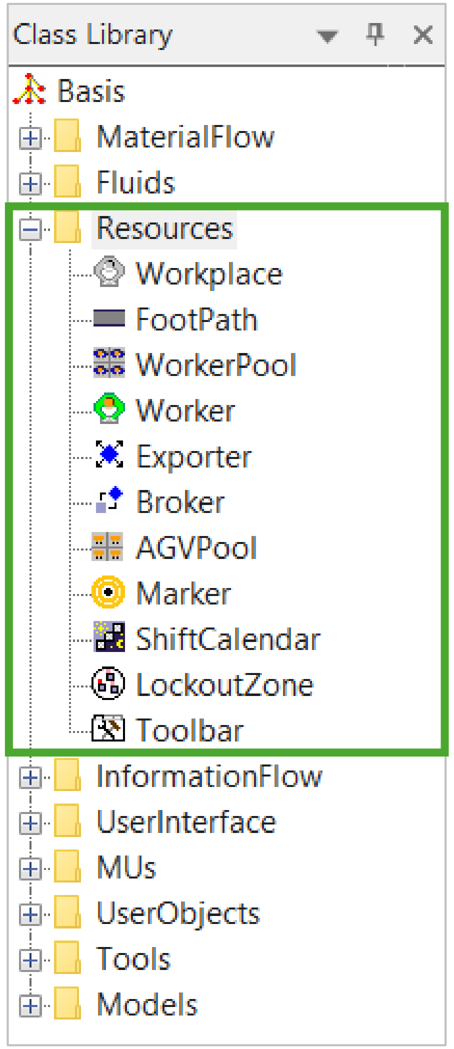
ŠKODA AUTO University

Resources

- The folder ”Resources” is designed for modeling of workers and their assigning to individual workplaces via ”WorkerPool” or to model automatically driven vehicles via ”AVGPool”.

<u>Elements of the resources:</u>

- Workplace
- FootPath
- WorkerPool
- Worker
- Exporter
- Broker
- AGVPool
- Marker
- ShiftCalendar
- LockoutZone



Plant Simulation Software Structure

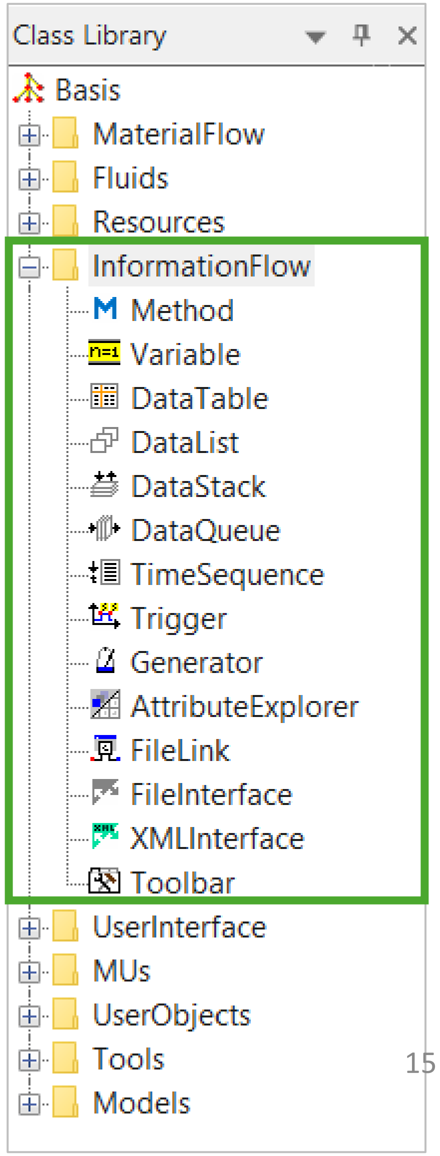
ŠKODA AUTO University

Information flow

- Exchange of information between the objects of the model can be ensured via objects of category ”Information Flow”.
- Object's behavior is possible to program by the object “Method” (when the defined conditions are met). The internal programming language “SimTalk” is used for this purpose (see in the next lectures).
- Communication with files of the type *.xls, *.xml, *.txt, etc. is possible through various interfaces.
- Files (queue, stack and card) and tables ensure the exchange of information among the objects.

Elements of the information flow:

- Method
- Variable
- DataTable
- DataList
- DataStack
- DataQueue
- TimeSequence
- Trigger
- Genrator
- AttributeExplorer
- FileLink
- FileInterface
- XMLInterface



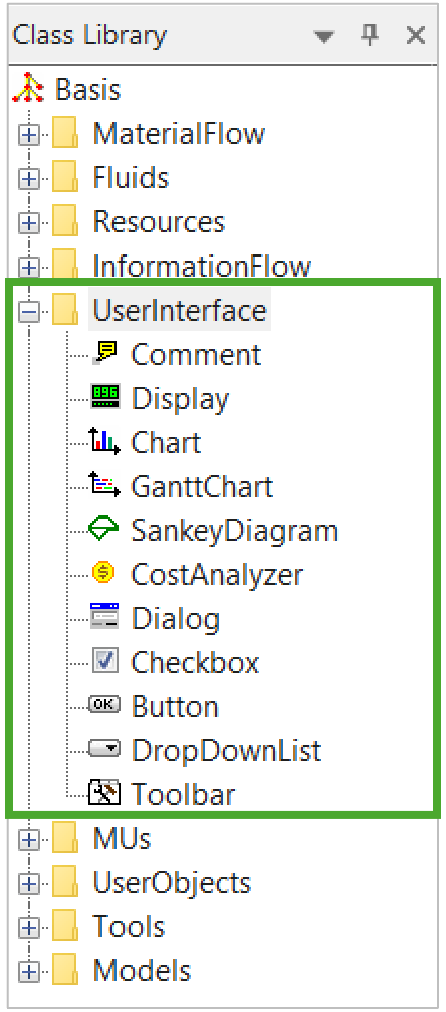
Plant Simulation Software Structure

User interface

- Items from the group ”UserInterface” is for comfort work with model inputs and outputs. User has a lot of functionalities available on the working area at that particular moment.
- Items are during the simulation active, in example graph ”Chart” is actualized during the simulation run.
- Using the buttons and programmed methods for them, it is possible to create, for example, a user-friendly environment for model parameterizing.
- The ”SenkeyDiagram” is used for more efficient evaluation of the material flow intensity in the model.

Elements of the user interface:

- Comment
- Display
- Chart
- Report
- SankeyDiagram
- CostAnalyzer
- Dialog
- Checkbox
- Button
- DropDownList

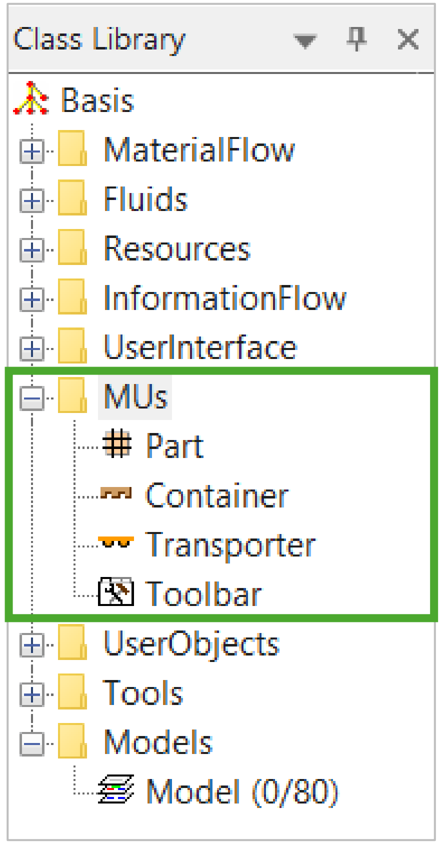


Plant Simulation Software Structure

ŠKODA AUTO University

Mobile units

- Mobile units "MUs" are not fixed on a place, unlike the material flow objects. They move along the material flow objects during the simulation thus they create discrete material flows. MUs move in jumps (due to the characteristics of discrete simulation).
- Mobile units have similar characteristics. They are different mainly in capability to carry other mobile units, prospectively capability to move itself.



Three basic types of MUs:

- Part — reworked subject (*product, goods*)
- Container — secondary transport medium (*pallet, container, preparation*)
- Transporter — vehicle (*truck, forklift*)

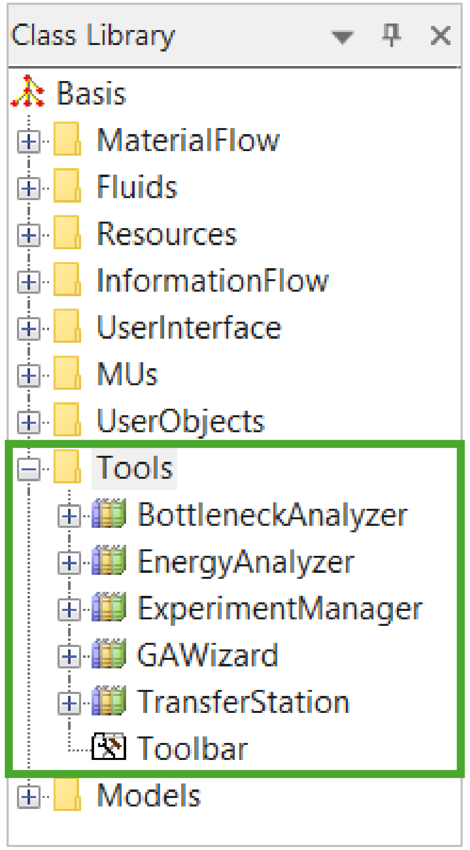
Plant Simulation Software Structure

Tools

- Folder "Tools" contains tools for easier work and evaluation of the simulation tests.
- "BottleneckAnalyzer" and "SekneyDiargam" are used for effective evaluation of the bottleneck places and material flow intensity in the model.
- "ExperimentManager" and "DistributedSimulation" support transparent and fast simulation tests.
- Using the experiment manager, we are able to define an experiment matrix that can be performed in a single loop.
- The "TransferStation" can be used to load, unload and move MUs.

Elements of the tools:

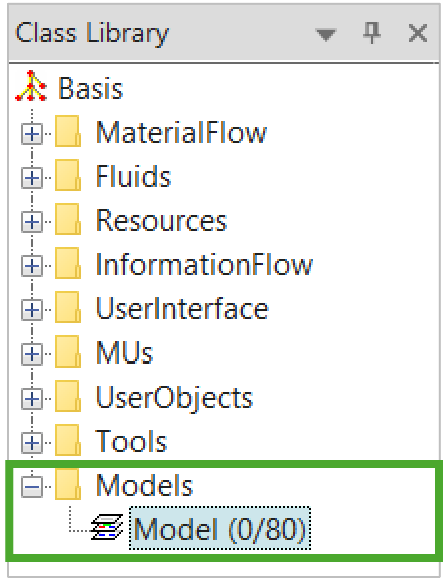
BottleneckAnalyser	analysis of bottleneck places
EnergyAnalyzer	energy consumption analysis
ExperimentManager	experiment manager
GAWizard	genetic algorithm
TransferStation	station transfer

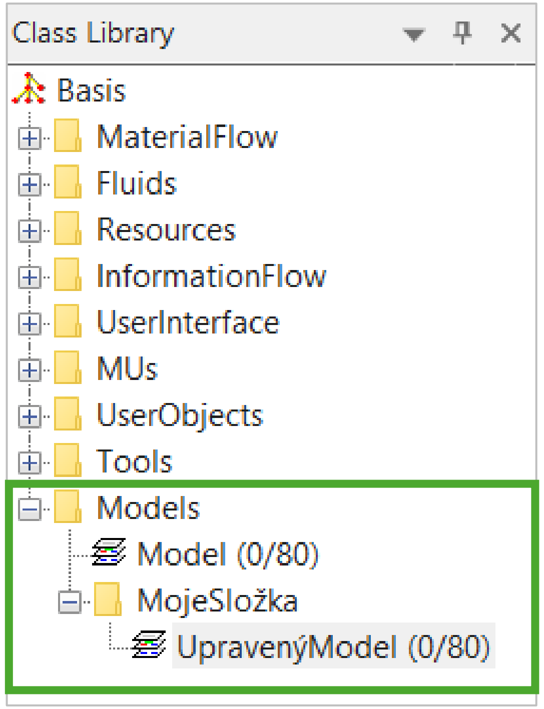


Plant Simulation Software Structure

Models

- Folder “Models” is aimed at saving of created models.
- The class library structure is possible to change, thus we are able to create models, model parts and tools according to the needs. It is necessary to have the class library structure clearly organized.
- Location of individual folders and objects in the class library can be changed by its marks and dragging with mouse on a different place in the same hierarchy level (“Drag&Drop”).
- It is necessary to hold “Shift” for change of the object location into different hierarchy level.
- WARNING!!! The additional changes in the structure of the folder names may lead to not functional current models because of the links on individual objects will not correspond to the new status.





Plant Simulation Software Structure

ŠKODA AUTO University

Name syntax

- Names must begin with a letter, afterwards letters or numbers can be, from characters the underscore “_” can be only used, i.e. “Working_station_1”, on the other hand “1_working_station” cannot be used.
- Names have not to be key words (e.g. if, then, else, from, until, loop, result etc.).
- Within the one name structure (i.e. in one frame) the same names cannot be used.
- There is no difference between lowercase and uppercase letters (i.e. in one frame the names “Station1” and ”STATION1” cannot be used – it leads to collisions).

Plant Simulation Software Structure

ŠKODA AUTO University

Principle of inheritance

- The principle of inheritance is a very advantageous feature of object-oriented systems, such as the Plant Simulation.
- Objects in the Class Library are classes, object templates, which are used for model creation.
- The principle of inheritance can be used during using of the class objects, because all the derived objects from a class inherit all its characteristics. If we change some attribute of the class object, the change will be automatically made on all derived objects of this class.
- Classes can be further modified; next derived classes can be also made. During this process, all attributes are inherited as well.
- The inheritance mechanism works on the one-way principle - it is inherited only from the class into derived objects, never the other way (only "top -> down").

Plant Simulation Software Structure

ŠKODA AUTO University

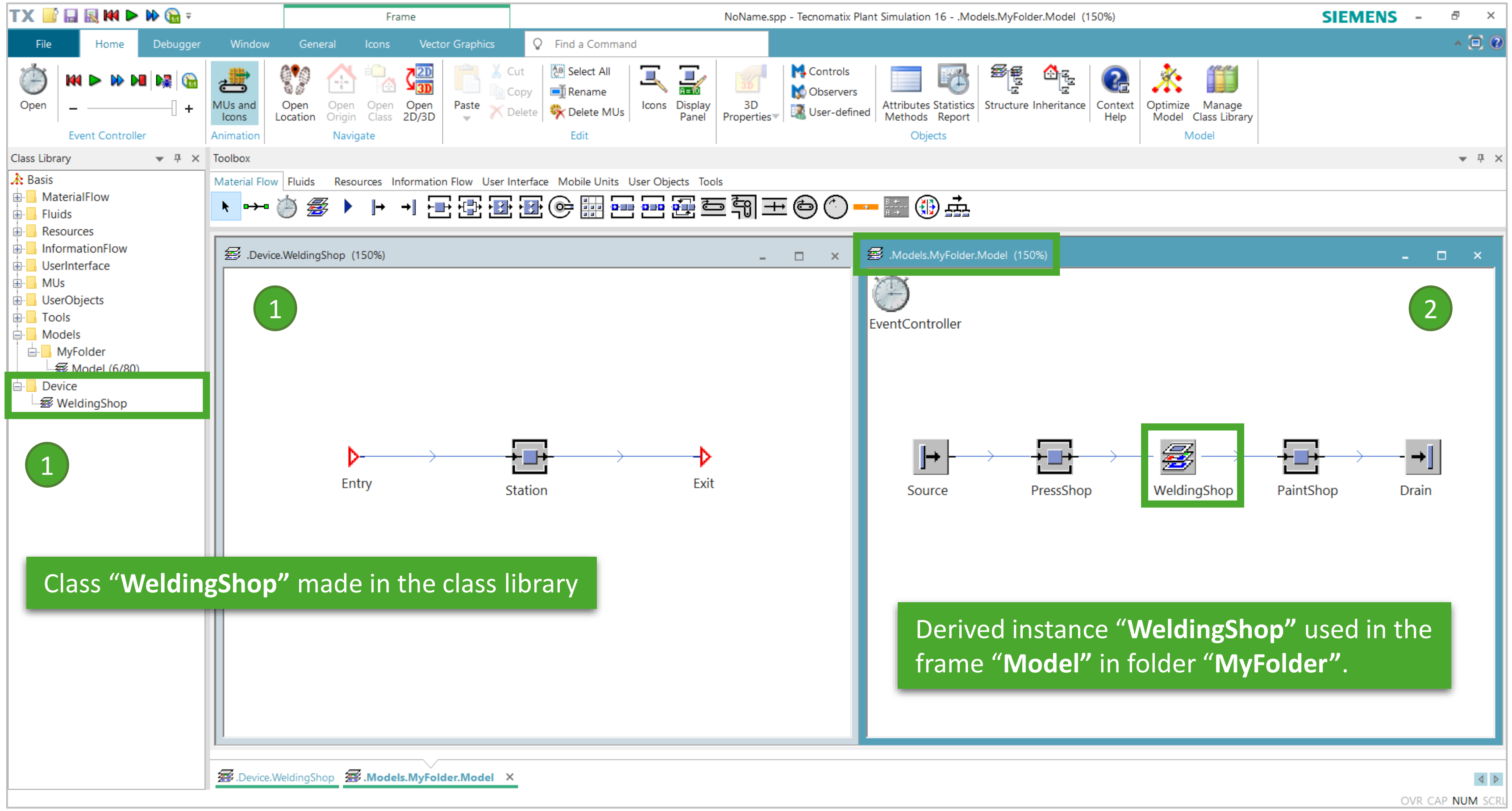
Principle of inheritance

- Class
  All the objects in the class library are marked as “classes”, they are as a template for instances that are inserted into the frames – e.g. machine.
- Subclass
  Object in the class library, which inherits all and the most important attributes from the class based on its structure. Other attributes are changed – e.g. different machines.
- Instance
  Instance is an object in the simulation model, which was derived from a class of library.
- Inheritance
  Class passes all its attributes to subclasses, or derived instances. This mechanism works only in the way from class to subclass/instance – not the opposite way!
- **All the changes, which are to be valid for all instances of one object, must be made in the object class!**

Plant Simulation Software Structure

Principle of inheritance

- The example of inheritance using "class" from class library on model creation by the "instance" form.

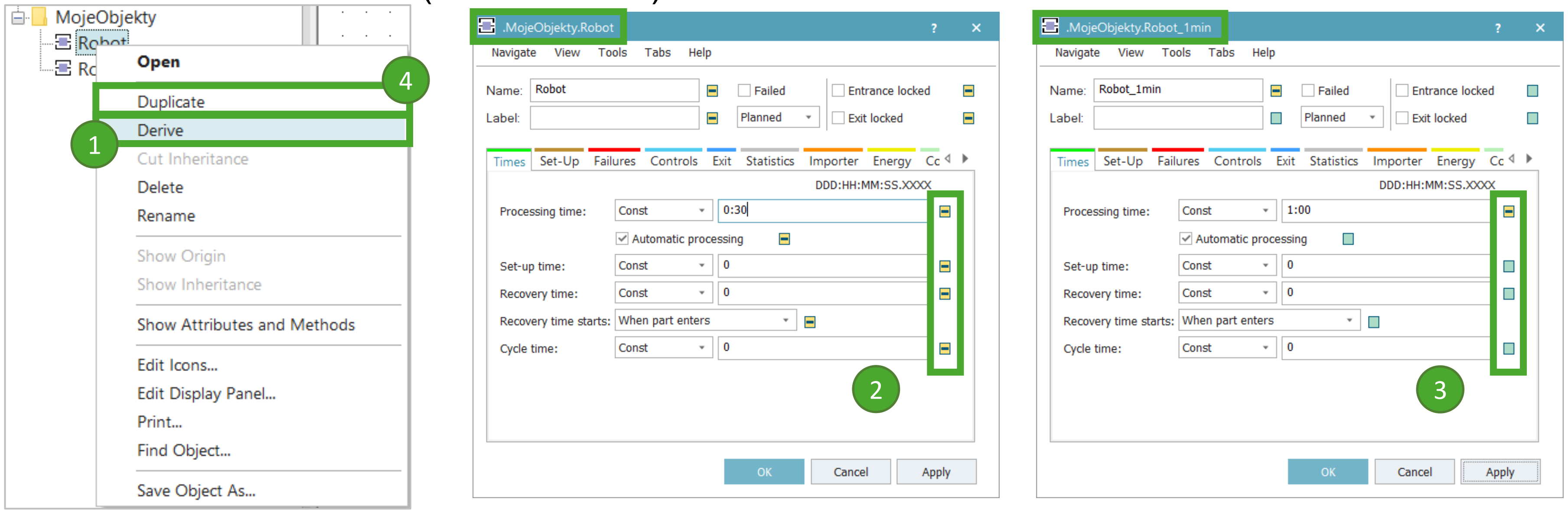


Plant Simulation Software Structure

ŠKODA AUTO University

Class, subclass

- With the command “Derive”(1) it is possible to create a subclass (3) from the class (2). This is useful, when we need to make several types of the same devices with, for example, different processing times on each of them (crossed out box). The inheritance is interrupted only for this attribute (the inheritance box is deactivated). The inheritance of the original is kept for the rest of attributes (class “Robot”).



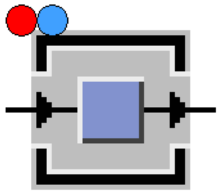
- With the command “Duplicate”(4) it is possible to make the identical clone from the class (2); without the connection to the original object. If we want to create “subclass” from this clone with the command “Derive”, the principle of inheritance between the clone and its subclass will be kept.

Plant Simulation Software Structure

ŠKODA AUTO University

Display of the object status using LED symbols

- Objects status of the material flow can be represented by LED-symbols on the upper edge of the object icon.
- It is possible to represent more situations at the moment:
  - Object status is in meaning of:
    - red — object has a failure - **failed**
    - blue — object is **paused**
    - green — object is **working**
    - yellow — object is **blocked**
    - brown — object is being **setting-up** (setUp)
    - light blue — object is **recovering** (no_entry)
    - orange — object is waiting for source

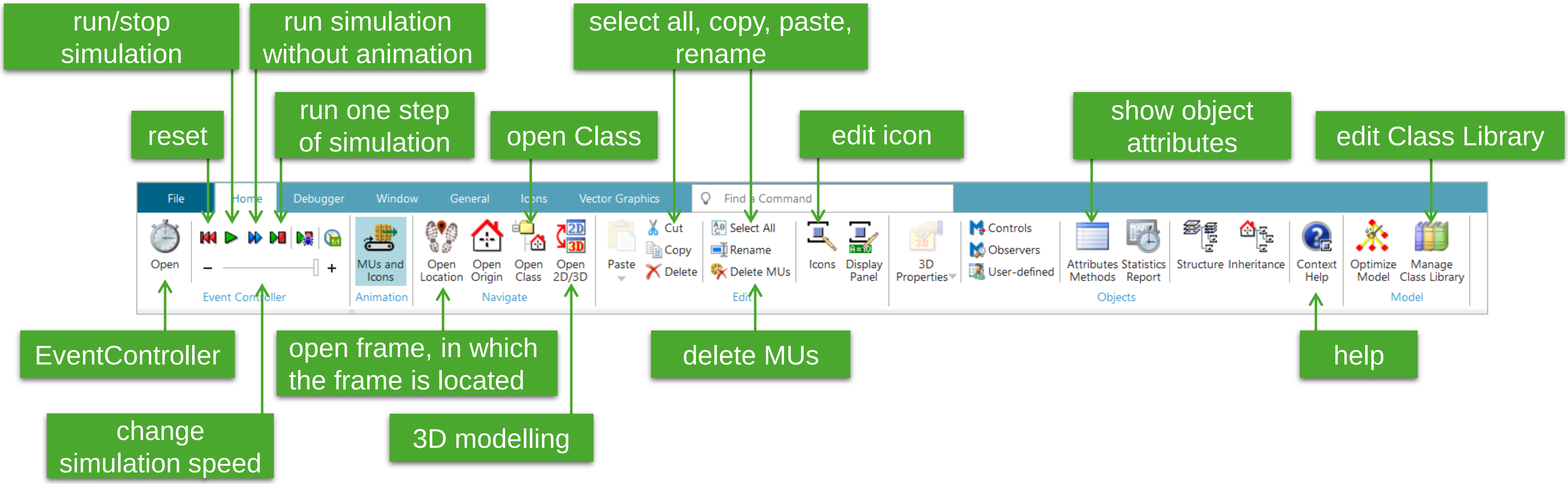


- Status “operational“– ready – does not have LED symbol.

Plant Simulation Software Structure

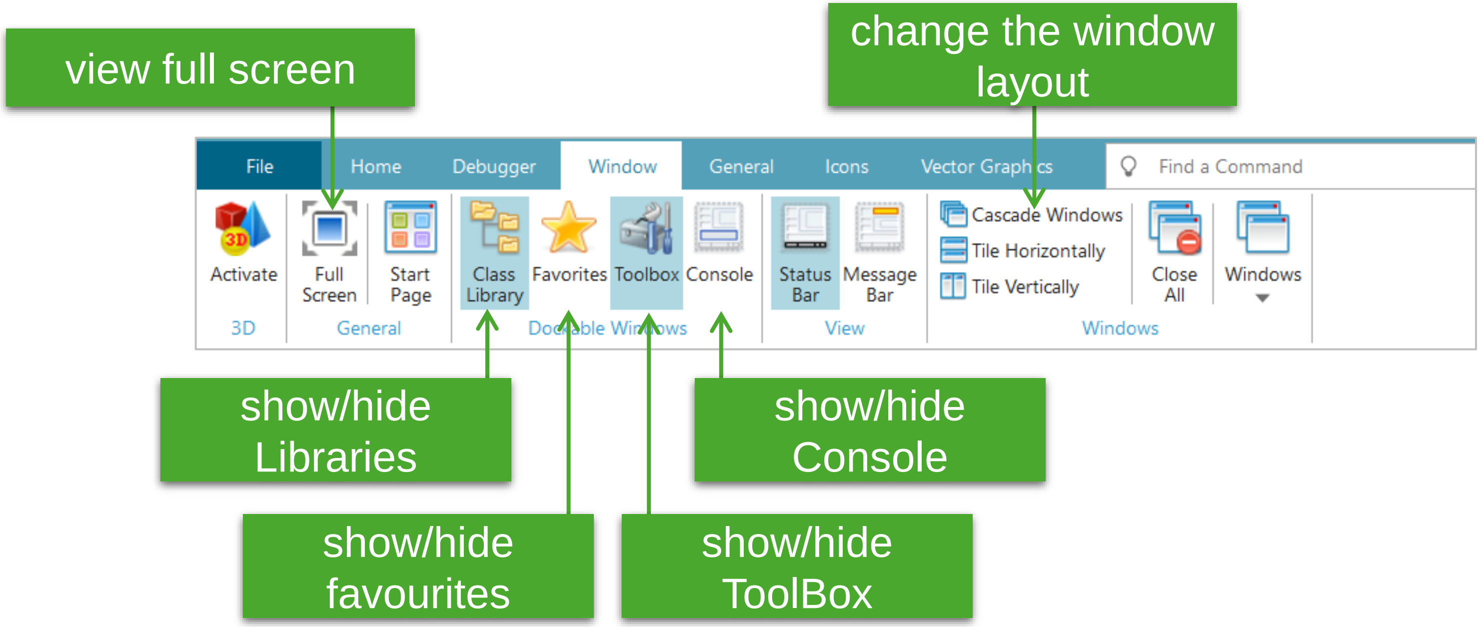
ŠKODA AUTO University

Menu bar in the panel "Home"



Plant Simulation Software Structure

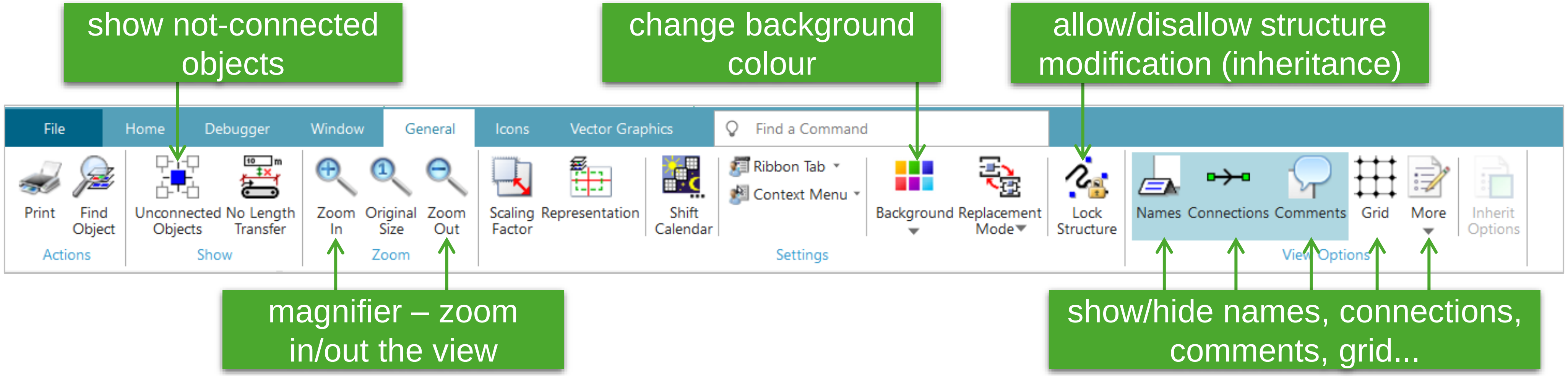
Menu bar in the panel “Window“



Plant Simulation Software Structure

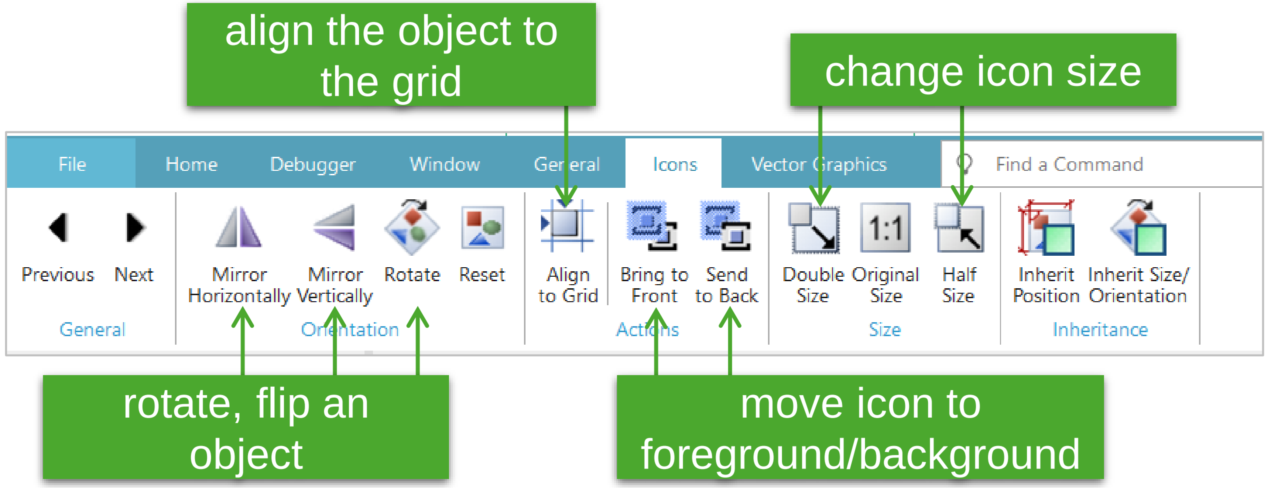
ŠKODA AUTO University

Menu bar in the panel “General“



Plant Simulation Software Structure

Menu bar in the panel “Icons“

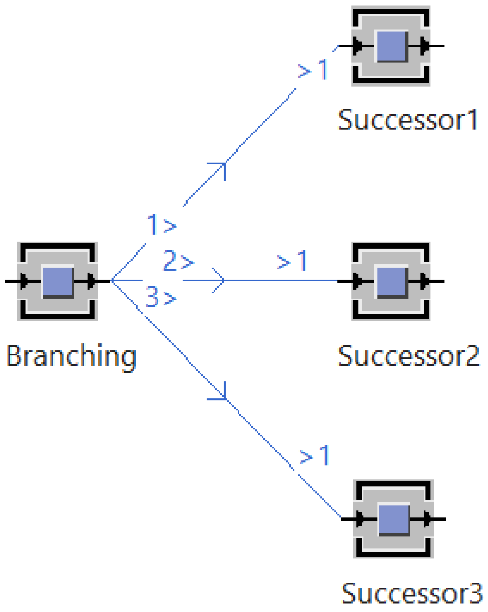


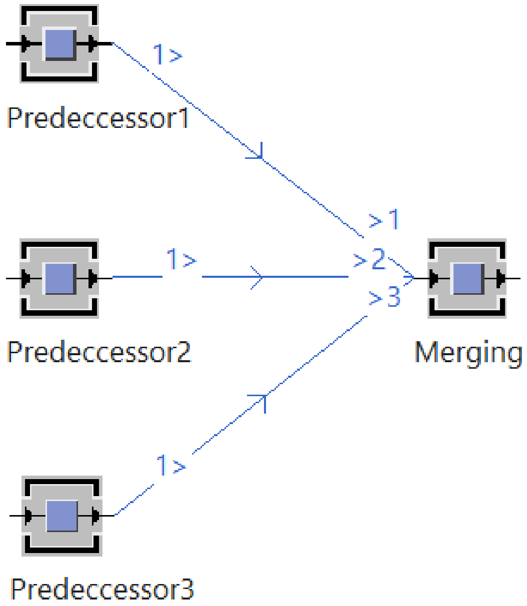
Plant Simulation Software Structure

ŠKODA AUTO University

Implemented strategies for moving of MUs from one object to another one

- The default behavior on the branching points is sequential/diverging. It means that MUs are passed on the following objects in the line.
- The FIFO rule – “first in first out” is applied to merging. If the next box is occupied, MU records its attempt to move at the next successor to the list of blocked MU from the beginning.



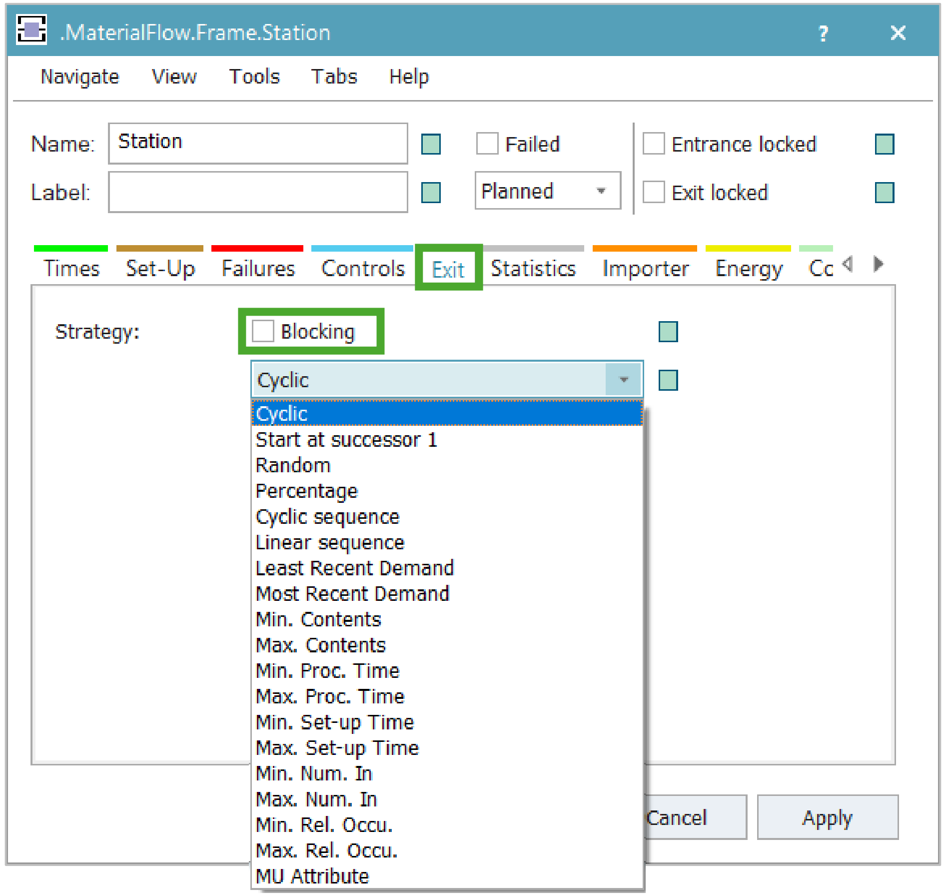


Plant Simulation Software Structure

ŠKODA AUTO University

Exit strategy from the object

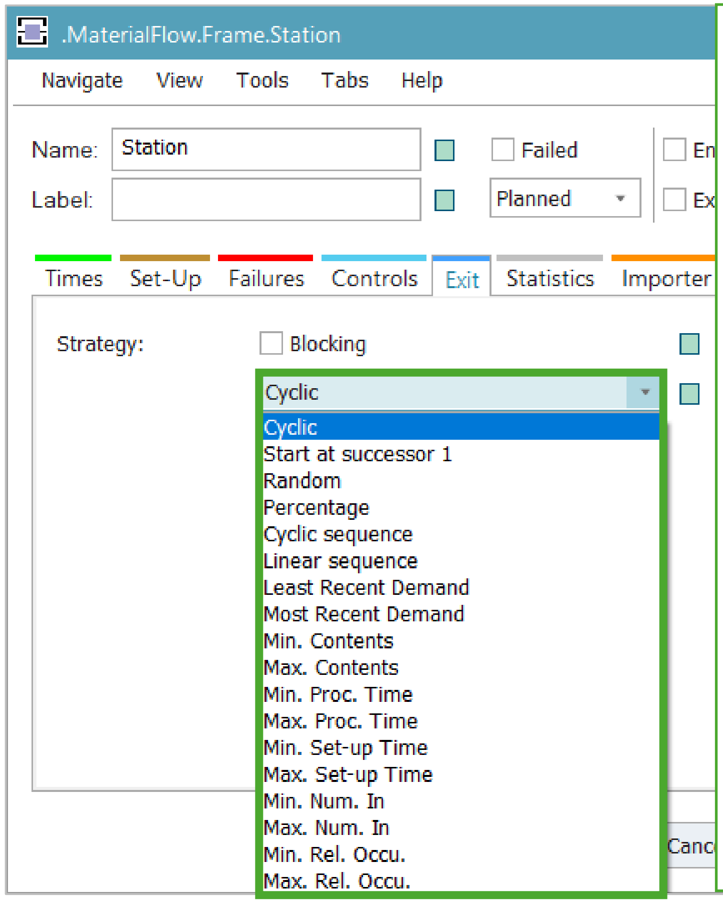
- The passing type or passing strategy of MUs to the following connected objects is set in the tab "Exit Strategy" for the material flow objects.
- Standard setting is the cyclic / sequential.
- The box "Blocking" determines, whether MU is forwarded or not. If the box is inactive and assigned MU's successor cannot accept it because it is occupied, the next successor in the order will be tested. If the box is active, the handover is awaited until this successor is able to accept MU.



Plant Simulation Software Structure

ŠKODA AUTO University

Output strategy from the object



- Cyclic – passing operates in the order of connected objects.
- Start at successor 1 – passing always starts at successor 1.
- Random – successor is randomly chosen.
- Percentage – allows percentage distribution.
- Cyclic sequence – the successor is chosen from the order determined by table; the next successor is specified by the following line during try to pass.
- Linear sequence – the successor is always chosen from the first line in the table.
- Least/Most Recent Demand – the oldest/the newest request, the successor is that object, which have waited on MU the longest/the shortest time.
- Other possibilities consider the settings or statistical parameters (for example the successor is the object with minimal / maximal content, minimal / maximal processing time etc.)

ŠKODA AUTO University

Thank you for attention

Jan Fábry

Department of Production, Logistics and Quality Management

fabry@savs.cz www.janfabry.cz

www.savs.cz