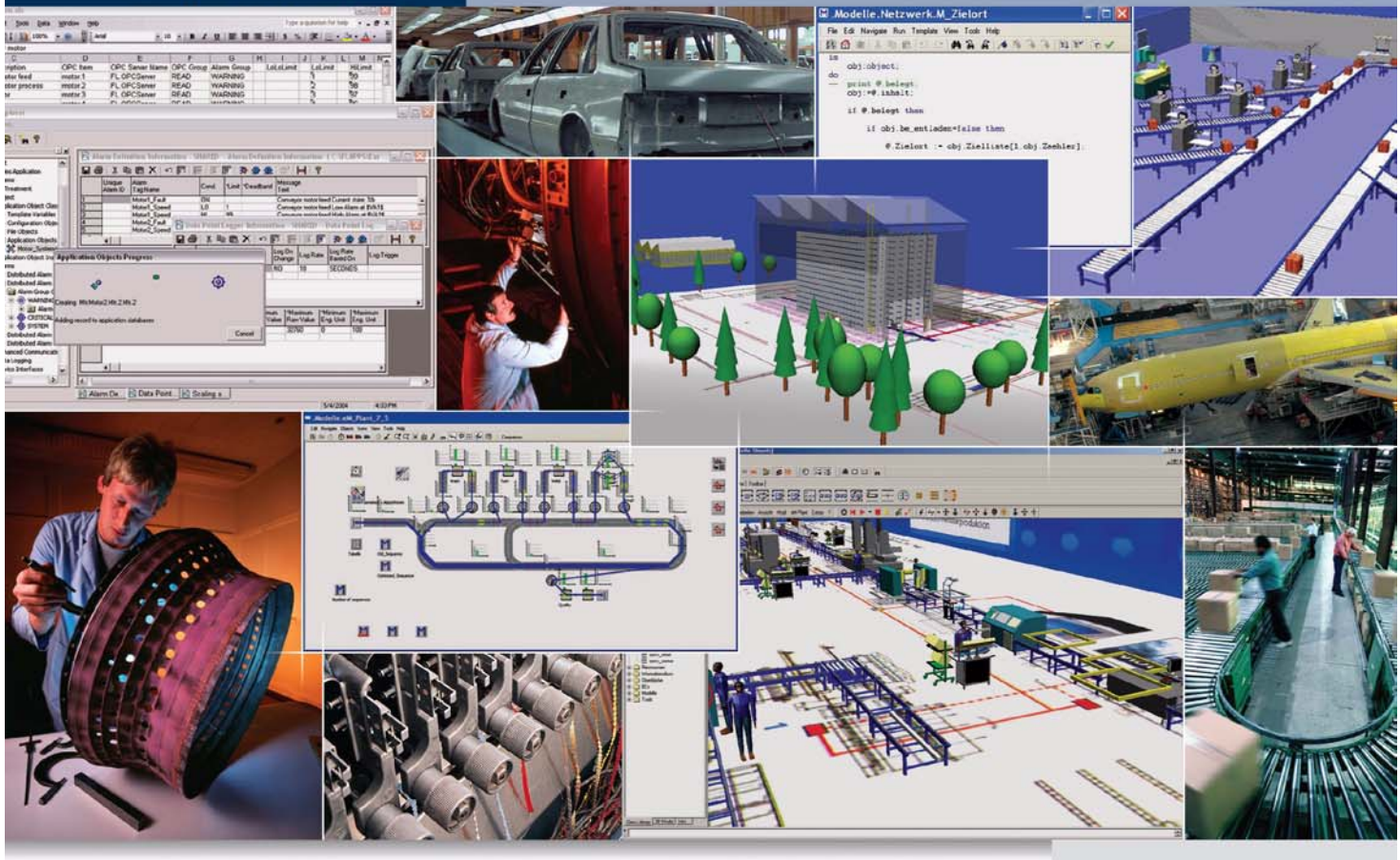
Plant Simulation SDX Interface

Reference manual

Siemens PLM Software

www.siemens.com/plm



TECNOMATIX

SIEMENS

Proprietary and Restricted Rights Notice

TECNOMATIX
Plant Simulation

SIEMENS

Plant Simulation SDX

November 2007

Table of Contents

Introduction

To evaluate and optimize factory layouts by running throughput simulation, Siemens PLM Software provides two products:

1. Factory Simulation Light
2. Plant Simulation-SDX

Factory Simulation Light is the application for the layout planning expert working with FactoryCAD. It's an optional product of FactoryCAD.

Plant Simulation-SDX is the application for the simulation expert working with Plant Simulation. It's an optional product of Plant Simulation.

The following document describes the usage of Plant Simulation-SDX. To find out how to enhance a layout in FactoryCAD to run simulations, please consult the **Factory Simulation** manual.

We recommend to attend an Plant Simulation basic and advanced training and a training for Plant Simulation-SDX before you start working with Plant Simulation-SDX.

Basic Workflow

To exchange data between FactoryCAD and Plant Simulation, we use the SDX data format. SDX stands for "**S**imulation **D**ata e**X**change" and is a standard for data exchange between layout design tools and simulation tools.

Consult the **Factory Simulation** manual to find out how to export SDX data from FactoryCAD.

To work with the SDX data in Plant Simulation we provide a file called SDX.spp. Loading this file in Plant Simulation, a dialog opens which guides you through the whole process.

Using this dialog

- You can import SDX files and build up a simulation model based on this data automatically.
- Then you usually will start working with the simulation model, e.g. adding scheduling information, loading additional order data from databases, implementing strategies or creating custom reports. Then you will start running simulations optimizing your system.
- When there's a new version of the layout available, import the new version of the SDX file in the same simulation model. Then you can use the model update functionality to update your simulation model without loosing all the changes you did in the model.

Getting Started

Start Plant Simulation SDX

It does not matter if you install Plant Simulation from the Tecnomatix 8.2 DVD or from the Plant Simulation 8.2 CD.

After installing Plant Simulation, install the latest patch for Plant Simulation from the GTAC download area (http://ftp.ugs.com). Note that you need your Webkey username and password for downloads.

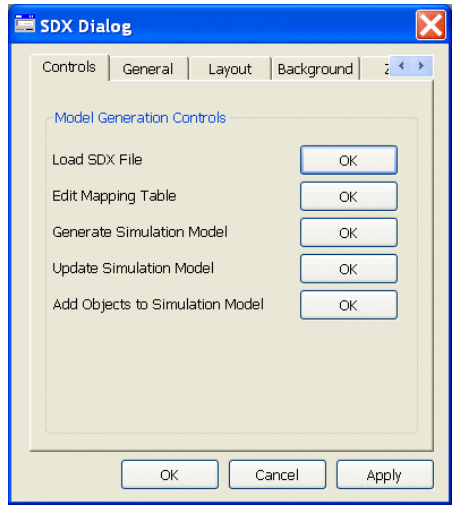
The software requires a carrier license for Plant Simulation (Plant Simulation Professional, Plant Simulation Standard etc.) and a license for the optional product Plant Simulation-SDX.

To start Plant Simulation-SDX

- Start Plant Simulation
- In the Explorer window on the left hand side, select the Info Pages tab

- Click on the link Libraries
- Click on the link SDX-Interface

Then Plant Simulation loads the integration model and the SDX dialog opens:



Picture 1:Controls tab of the SDX dialog

From this dialog, you can load SDX files, edit the mapping of FactoryCAD objects to Plant Simulation objects, generate simulation models or update simulation models.

Settings

Note: Note that most settings described below will take effect only for the next set of SDX data you load or after a reload of the SDX integration model.

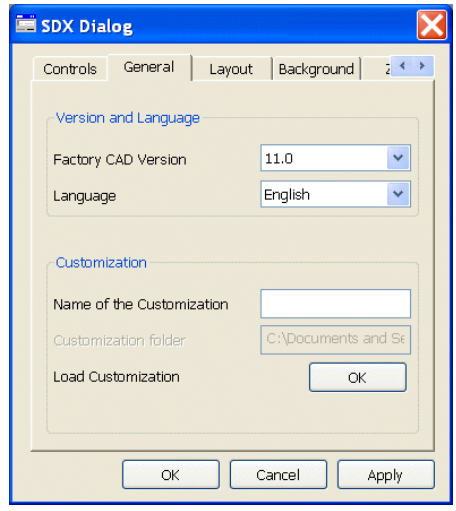
General Tab

FactoryCAD Version

If FactoryCAD is installed on the same machine, select the version of FactoryCAD.

Language

Select the language of the SDX dialog, messages and result reports.



Picture 2:General tab of the SDX dialog

Layout Tab

Frame Size for New Models

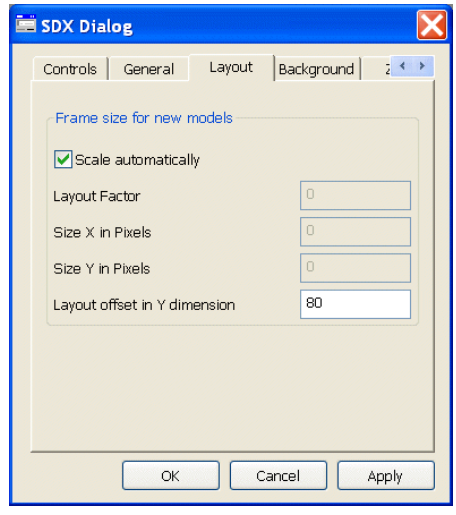
Use this tab to specify the size of the Plant Simulation frame which the SDX dialog will create to generate a new simulation model.

If you check Scale automatically (recommended), the SDX dialog calculates the size automatically, based on the extensions of the drawing. If you uncheck this item, you can set

1. the Layout factor which will calculate the size of the frame by multiplying the extensions of the drawing with this factor. Common values are values of 5 to 20
2. the Size x in pixels/Size y in pixels so the SDX dialog will work with a fixed size.

Note, that the Size x in pixels/Size y in pixels overwrites the Layout factor.

The Layout offset in y dimension reserves a certain area on top of the frame where the SDX dialog will not place any objects. This area is useful for administration and control objects, like the event controller, certain methods, reports etc.



Picture 3:Layout tab of the SDX dialog

Background Tab

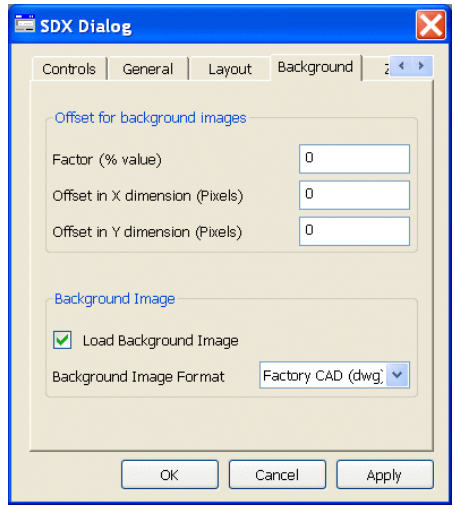
Background Image

If you would like to load the FactoryCAD drawing as a background image of the simulation model, check Load background image.

Offset for Background Images

If the image does not fit well, you can change the scaling of the background image by entering a factor. E.g. a factor of "10" will increase the size of the background image by 10%. A factor of "-10" will decrease the size of the background image by 10%.

Entering a value under Offset in x dimension or Offset in y dimension, you can move the background image of the drawing.



Picture 4:Background tab of the SDX dialog

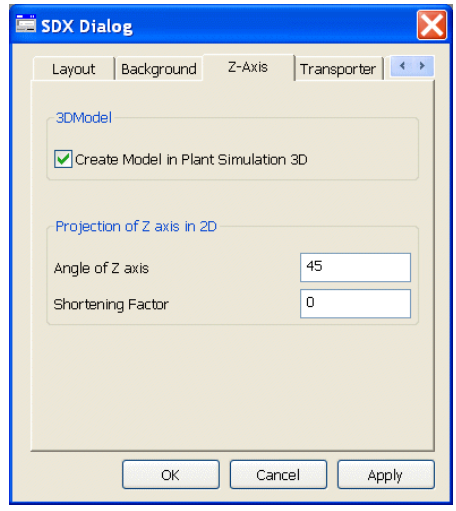
Z-Axis Tab

3D Model

Check Create model in Plant Simulation 3D to activate model generation in Plant Simulation 3D.

Projection of Z Axis

By default, objects on different floors in the layout will appear above each other in the 2D view of the simulation model, the 2D view provides a view from the top. If you enter a Shortening factor, Plant Simulation moves objects in the specified angle to the top and to the right depending on the z coordinate (height of the reference point) of the object, providing a projection of the z axis in the x/y plane.



Picture 5:Z-Axis tab of the SDX dialog

Transporter Tab

Generation of Transporters

If transporters should wait at certain stations for parts ("home stations"), check the corresponding fields.

EMS transporters will wait at load stations.

For AGVs, Fork trucks etc., specify the Home station type where transporters should wait. In this case the quantity of transporters available will be assigned equally to the quantity of stations of the specified type. So if the transporter has no order, it will return to it's home station and wait there for new orders.

Note: If you define that transporters should wait at certain stations, make sure that these stations are not located on the main road, because if transporters wait at this station, they might block other transporters which would like to pass.

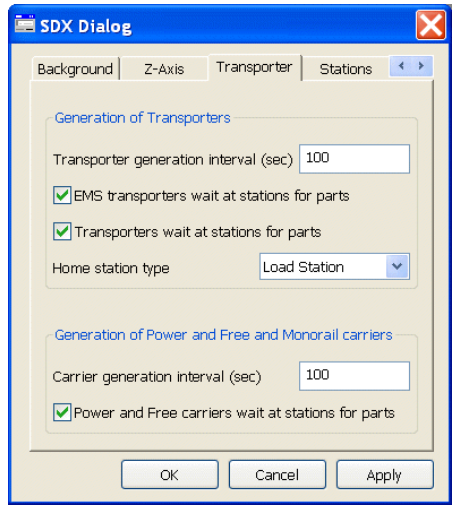
Note: If there's more than one transporter assigned to one station, then it might happen that two transporters wait at this station. Now if the second transporter get's an order, it might not be able to fulfill this order, because it's blocked by the first one. Therefore it's useful to define as many home stations as you have transporters in the system.

When the simulation starts, the system is empty. Use the Transporter generation interval to specify the interarrival rate of transporters in the system.

Generation of PowerAndFree and Monorail Carriers

If carriers should wait at load stations for parts, check the corresponding field.

When the simulation starts, the system is empty. Use the Carrier generation interval to specify the interarrival rate of carriers in the system.

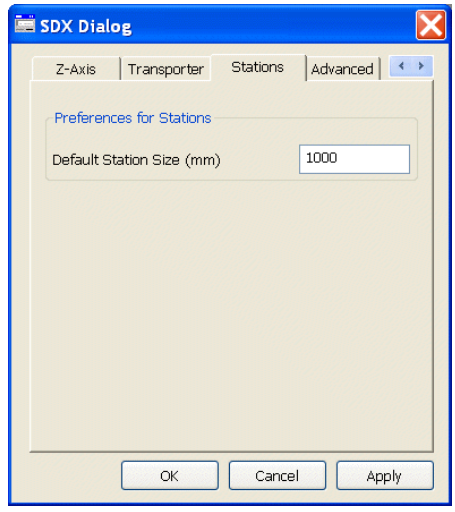


Picture 6:Transporter tab of the SDX dialog

Stations Tab

Preferences for Stations

In FactoryCAD, stations on transport system or aisles are just points in the layout, without extension. For the simulation, a station needs a certain extension. Use the Default station size to specify the default extension. The length unit is the unit of the last SDX file you imported.



Picture 7:Stations tab of the SDX dialog

Steps of a Model Generation Process

The automatic model generation from SDX data happens in four steps:

1. In a first step, the SDX dialog imports the SDX file as it is in an Plant Simulation table
2. In a second step, Plant Simulation copies the data to different tables depending on the object type and calculates type specific data.

3. In a third step, Plant Simulation generates a Creation Table which contains the relevant information for all objects to be generated in the Plant Simulation model. During the creation of this table, the SDX dialog converts the data to Plant Simulation units, converts layout coordinates etc.
4. In the last step, Plant Simulation inserts the objects in the simulation model and sets the parameters of those objects, based on the data in the Creation Table.

The steps 1 and 2 happen when you click on Load SDX file on the Controls tab of the SDX dialog.

The steps 3 and 4 happen when you click on Generate simulation model on the Controls tab of the SDX dialog.

Import SDX Files

On the Controls tab of the SDX dialog, click on Load SDX file. A dialog opens, where you can select the file to import.

Note: The default extension of SDX files exported from FactoryCAD is *.sdx. Even though the data format is SDX, the integration supports only the file format XML (which is also the default file format to export SDX data from FactoryCAD).

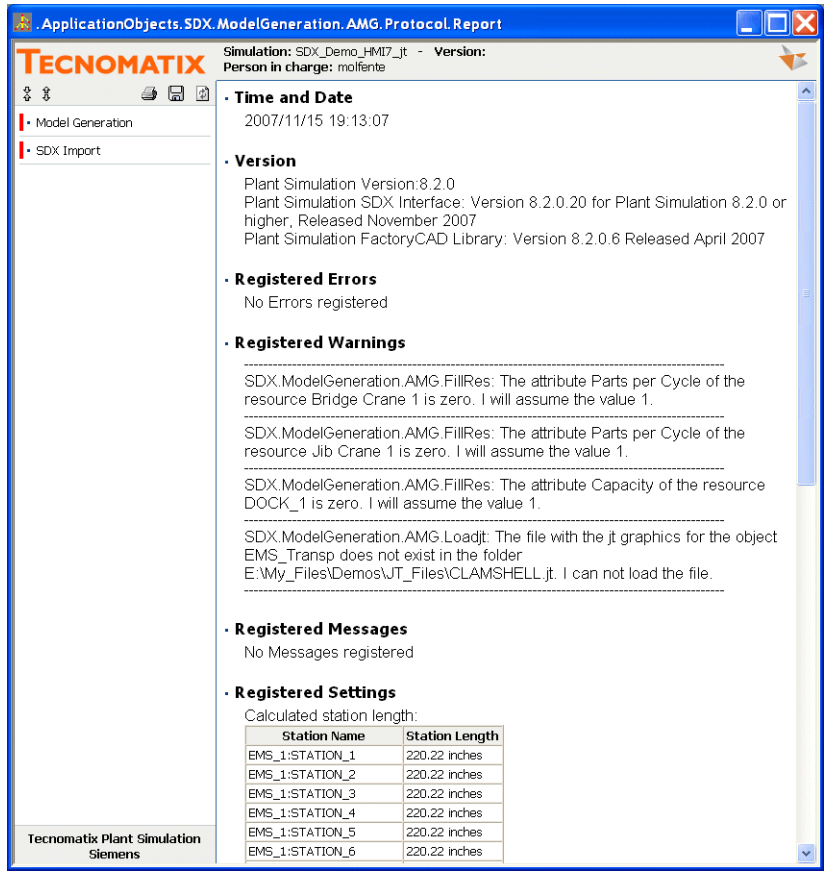
Note: There is a versioning for SDX, too. The current version of the integration supports SDX version 4.4. This is the default version to export SDX data from FactoryCAD.

Generate Simulation Models

On the Controls tab of the SDX dialog, click on Generate simulation model.

The Loading Report

At the end of the model generation process, the SDX dialog opens a loading report.



Picture 8:Loading Report of the SDX dialog

The report shows time and date and the versions of Plant Simulation, the SDX integration model and the library of FactoryCAD objects used in Plant Simulation for the automatic model generation.

Then the report contains 4 types of messages: Errors, Warnings, Messages and Settings.

1. Errors

   If the SDX file is missing information (e.g. no products are defined) or contains wrong information (e.g. two objects with the same name), these errors are listed in this section. If the simulation model classifies problems as errors, it means that the simulation can not start unless the problems are fixed.

2. Warnings

   If the SDX file contains any (potential) problems for the simulation, they are listed as a warning. Depending on the type of warning, the simulation might return wrong results. The SDX dialog will assume reasonable values, if possible.

3. Messages

   Information listed in the messages section are just for your information, they will not lead to wrong simulation results.

4. Settings

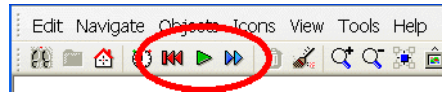
   If the simulation model calculates values from the SDX file (e.g. the length of stations), this information is listed here.

Running Simulations

Start/Stop the Simulation

To Start/Stop a simulation run

1. Click the Reset icon in the toolbar of the simulation model
2. Click the Start/Stop icon in the toolbar of the simulation model. If you know, the simulation runs the expected way, use the Start without Animation icon to perform simulation runs faster



Picture 9:Start/Stop a simulation run

To control the speed of the simulation, open the EventController in the toolbar of the simulation model.

Picture 10:Open the event controller

Executing Multiple Simulation Runs

Open the dialog of the Results object by double click.

Picture 11:Results object

Why You Would Like to Execute Multiple Simulation Runs

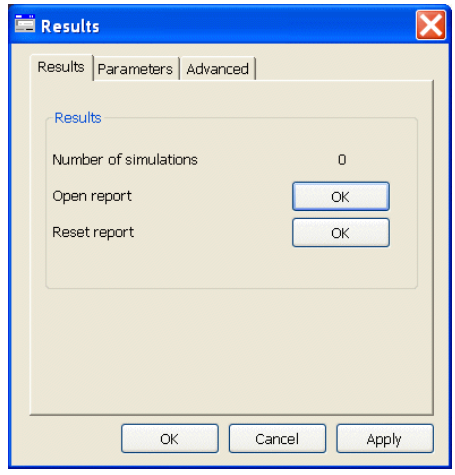
Several parameters in your simulation model depend on random behavior, e.g. the breakdown behavior of stations, machines or robots, in some cases even the cycle time of stations if manual work is involved. Therefore, two simulation runs with different random numbers will lead to different results. But to take a decision over a multi million dollar system, you would like to know which results you can expect from the real system.

The solution for this problem is to execute multiple simulation runs with different random numbers (“observations“). Then the results report will show the results for each simulation run and the average results. The list of results for each simulation run shows you the “bandwidth” of results and is a valuable indicator for the stability of your system. The average value is the value you can expect from the real system.

The Results Tab of the Results Object

The Results tab shows you the number of simulations you already executed for this simulation model.

Click the Open report button to open the html report which shows the results of the last simulation runs. To delete the results of the last simulation runs from the report, click the Reset report button.



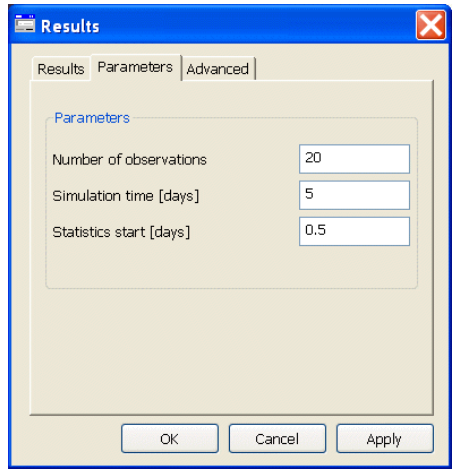
Picture 12:Results tab of the Results object

The Parameters Tab of the Results Object

To execute multiple simulation runs for the same simulation model using different random numbers ("Observations"), specify the Number of observations.

Specify the number of days you would like to simulate under Simulation time.

Specify the start of the statistics collection under Statistics start. Since the simulation model is empty at the beginning of the simulation, the purpose of the Statistics start is to start statistics collection only after the ramp-up phase.



Picture 13:Parameters tab of the Results object

Note: When you change settings of the results object, remember to reset the simulation before you start a new simulation.

Open Charts

In the top area of the simulation model, you find several evaluation tools. To open charts during the simulation run

1. Right-click the buffer histogram and choose Show Display Window from the context menu.

Picture 14:Buffer Histogram

2. Right-click the charts for machines, operators, stations or transporters and choose Show Display Window from the context menu.

Picture 15:Charts for machines, operators, stations and transporters

3. Right-click the Bottleneck Analyzer and choose Analyze from the context menu.

Picture 16:Bottleneck Analyzer

4. Right-click the Sankey Diagram and choose Display Sankey from the context menu.

Picture 17:Sankey Diagram

5. After the first simulation run, you can always open the html report again to look at the last results. To open the report, right-click the Report object and choose Show Display Window from the context menu.

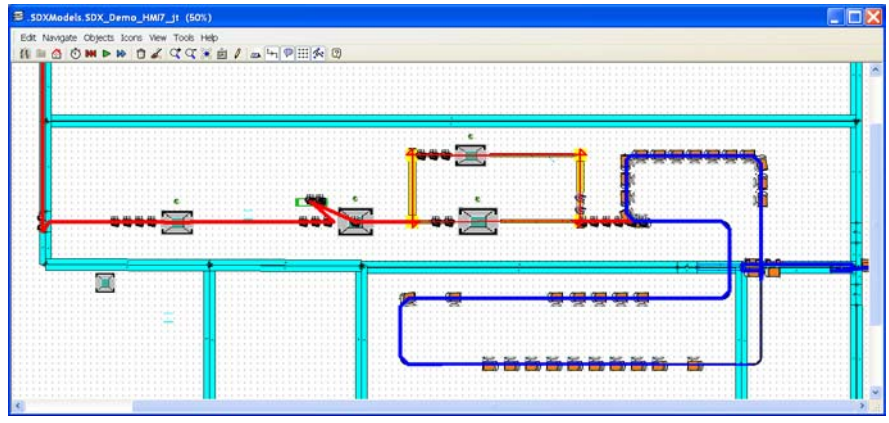
Picture 18:Html Report

The Results Report

After a complete simulation run, the result report opens. To look at a certain page of the report, click on the pages in the tree on the left hand side of the report window under Result Values:

1. Sankey

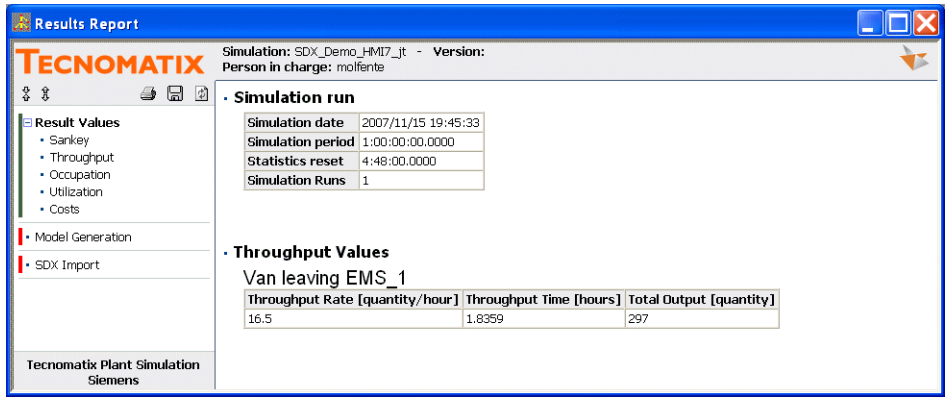
   This page shows a screenshot of your simulation model including a sankey diagram. The thickness of the red line of the sankey diagram shows how many parts per hour went a certain way.



Picture 19:Sankey diagram in the results report

2. Throughput

This page shows you how many parts per hour left the system (Throughput Rate), how long the parts were in the system (Throughput Time) and how many parts left the system during the simulation time you entered.(Total Output).

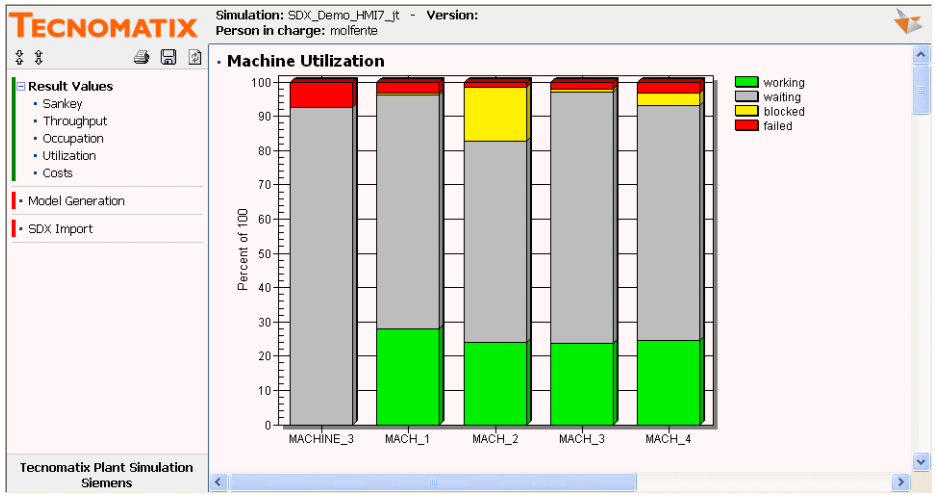


Picture 20:Throughput values in the results report

3. Utilization

This page shows

- Charts of the utilization of machines, operators and stations



Picture 21:Utilization charts in the results report

The chart shows how many percent of the simulation time a resource was working, waiting for material, blocked by successor resources or disrupted (compare the legend on the right hand side).

Note: The chart shows only stations which are relevant for the production process, i.e. stations with the internal type PROCESS RUN or PROCESS STOP station.

- Charts of the utilization of transporters

  The charts shows how many percent of the simulation time a transporter was in which state. The charts show the following states:

Occupied - with order: Products are loaded on the transporter and the transporter has an order to bring the products to a target station. Note that the time to load and unload the transporter is also part of this time.

Empty - with order: The transporter is empty but it was reserved by a station. So it is on it's way to this station to pick up products.

Empty - driving home: The transporter is empty and on it's way to it's home station (the place where the tranporter waits when it has nothing to do). Note that home station are available only when you checked Transporters wait at stations for parts on the Transporters tab of the SDX dialog.

Empty - without order: The transporter is empty and has nothing to do (waiting).

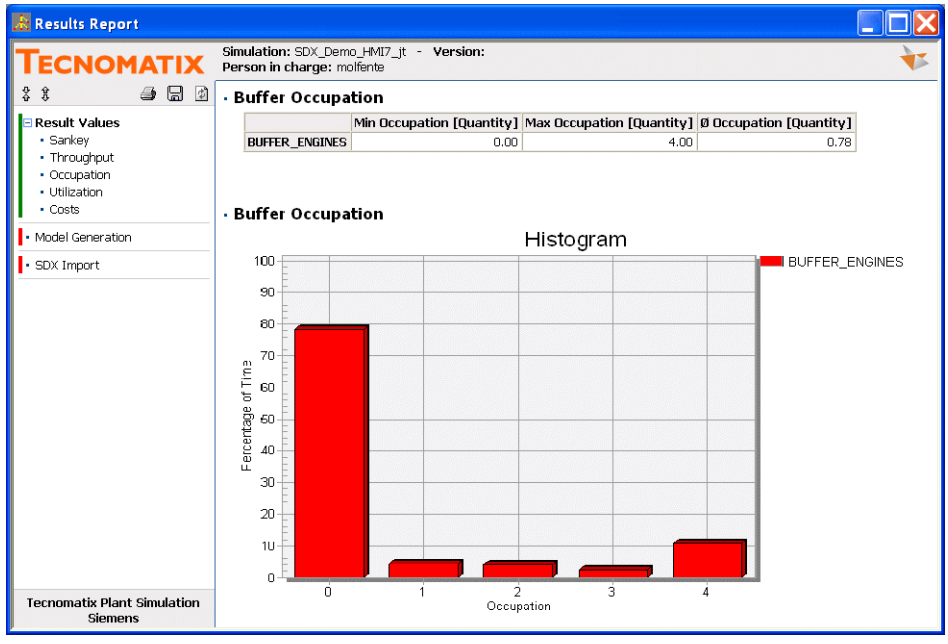
Failed: The transporter has a breakdown.

4. Occupation

This page shows

- A table which lists the minimum, average and maximum occupation of buffers in your system.
- A histogram of the occupation of buffers.

  The histogram shows how many percent of the simulation time how many places of the buffer were occupied.



Picture 22:Occupation page of the results report

5. Costs

This page shows the total costs of the system. These costs consist of

- The sum of all invest costs
- Maintenance and repair costs per day based on the actual maintenance/repair activities during the simulation
- Labor costs per day based on the actual work times during the simulation
- Fuel/Power costs per day based on the actual operating time during the simulation

Storage of Result Reports

HTML Report

The SDX dialog saves all result reports automatically in a folder containing HTML files in the same folder where you saved the SDX file. The name of this folder is: "<SDX filename> Report <Date>_Files". So if the name of your SDX file is "Test.sdx" and the simulation date is June 3rd 2008, the name of the folder is "Test Report 2008-06-03_Files".

You can open the report with any web browser. To open the report in this folder, double click the file "_Start.htm" in this folder.

EXCEL Export of Simulation Results

The SDX dialog saves the key result values automatically as an EXCEL file in the same folder where you saved the SDX file. The name of this file is: "<SDX filename> Report <Date>.xls". So if the name of your SDX file is "Test.sdx" and the simulation date is June 3rd 2008, the name of the folder is "Test Report 2008-06-03.xls".

Update Simulation Models

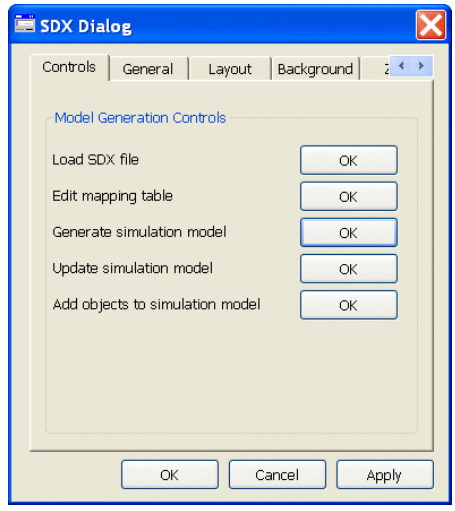
Execute Model Update

Note: Always reset the simulation model before you execute a model update.

To update an existing simulation model with a new version of the layout data, perform the following steps:

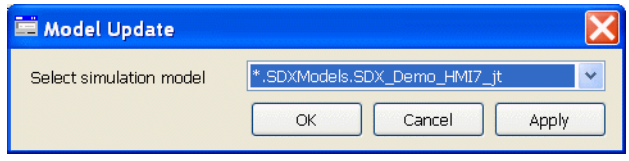
1. Import the SDX data which contains the new layout data using the Load SDX file button on the Controls tab of the SDX dialog.
2. There are two buttons to update the simulation model: Update simulation model and Add objects to simulation model.

   Click on Update simulation model in case the new SDX file contains data of the same layout area you already imported. Click on Add objects to simulation model in case the SDX file contains data of an additional area of the layout, e.g. if you ran simulations only for one production line first and now you would like to simulate two production lines in parallel (so the new file contains only the data from the second line). In case the new file contains data from the area you already simulated AND data from new areas, use Update simulation model.



Picture 23:The Controls tab of the SDX dialog

Then a dialog opens where you can select the simulation model to update, in case you have more then one simulation model (e.g. to run simulations for more than one area or for different alternatives of the same area) in your Plant Simulation model.



Picture 24:Select simulation model to update

At the end of the update process, a report opens which lists all the changes in the SDX data in the section Registered Messages.

The Data Object

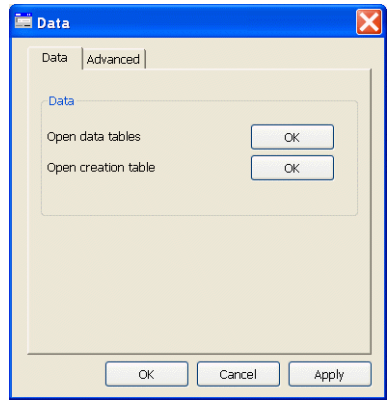
To be able to update simulation models with a new version of the layout data, the SDX dialog saves all the data used to build up the simulation model. To view this data, open the Data object

Picture 25:The Data object

The Data Tab

Click on Open data tables to open the frame which contains the SDX data in the original units.

Click on Open creation table to open the Creation Table which contains all the data of the objects in the simulation model in Plant Simulation units.



Picture 26:The Data tab of the Data object

Customization

Note: Before you start customizing the SDX.spp file, we recommend to save the file in a different folder or a different name and to keep the original SDX.spp file in the original folder. This way you can use the original file again e.g. if you start developing a new customization for a different purpose.

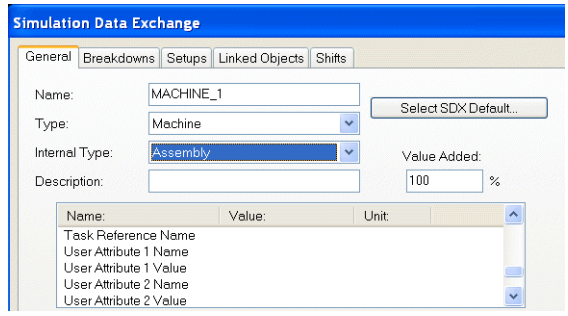
Use Custom Objects for the Model Generation

Custom Objects in the SDX Data

The mapping of SDX objects to Plant Simulation objects is based on the attribute INTERNAL_TYPE of SDX objects.

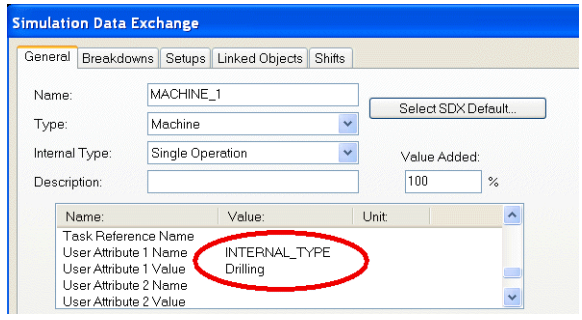
In FactoryCAD, set the internal type attribute of an object based on the type of object:

1. Use the dialog of FactoryCAD objects from the FactoryCAD toolbars to set the internal type.



Picture 27:Internal type attribute of FactoryCAD objects

In case, you would like to work with additional internal types, you can set the internal type using the custom attributes of the FactoryCAD object



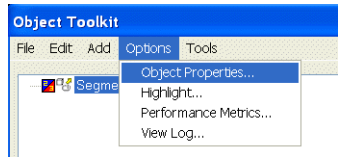
Picture 28:Internal type as custom attribute of FactoryCAD objects

Note: When you define a custom attribute INTERNAL_TYPE, the value of this attribute will overwrite the default value i.e. the value displayed in the dialog under Internal Type.

Note: Note that the attribute name INTERNAL_TYPE and the attribute value are case sensitive.

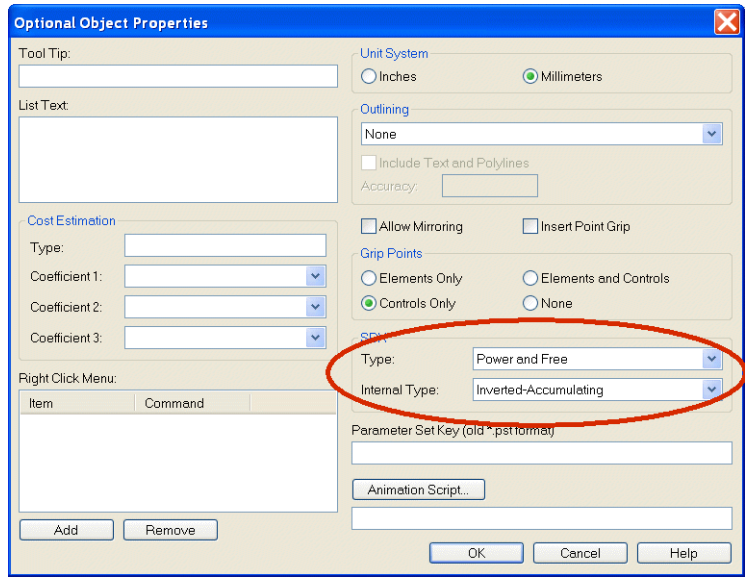
Note: Note that AutoCAD graphic elements (“blocks”) don’t have SDX attributes. If there are AutoCAD blocks which are simulation relevant, convert those elements to FactoryCAD objects first using the icon Replace block with generic tool from the Industrial Objects toolbar. Then you can set the SDX type and internal type like for all FactoryCAD objects.

2. When you use the XML Object Toolkit in FactoryCAD to create your own objects, it’s important to define the SDX type of object, so that Plant Simulation knows which Plant Simulation object to use for the model generation. To set the SDX type of a custom XML object in FactoryCAD, select Options > Object Properties from the menu of the XML Object Toolkit.



Picture 29:Object properties in the XML object toolkit

Then you can set the type and internal type of the object.



Picture 30:SDX type and internal type of custom XML objects in FactoryCAD

Using Custom Plant Simulation Objects for Model Generation

There are two possibilities how to create custom Plant Simulation objects which can be used for the automatic model generation:

1. Derive objects from objects in the FactoryCAD library
2. Use objects from other libraries or created from basic objects

Derive Objects from Objects in the FactoryCAD Library

All the objects from the FactoryCAD library designed for this integration communicate with each other in a certain way. If you derive objects from existing objects from this library, those objects will continue to use this communication workflow. This enables the model generation/model update algorithms to fill the operation plan and predecessor/successor relationship automatically from the SDX data.

To change the behavior of derived objects, you have three possibilities:

1. Change attributes of the object in the dialog of the object.
2. Objects from this library contain user methods where you can design your own special behavior of those objects. These methods are described in detail in the chapter Working with the Model below.
3. Open the Plant Simulation object from the menu Navigation > Open Object to add or modify additional attributes.

Use Objects from Other Libraries or Created from Basic Objects

You can also use basic objects, objects from Plant Simulation application object libraries or your own custom Plant Simulation objects (objects from your own libraries). In this case the SDX dialog can provide only the following functionality:

1. During model generation, the SDX dialog can insert these objects in the correct position in the layout and zoom the object icon to the appropriate size (using the layout factor for the frame).
2. The SDX dialog can write attribute values from the SDX data to the object as specified in the mapping table, see the chapter Map SDX Objects to Plant Simulation Objects below.
3. The SDX dialog can connect objects with predecessors/successors. In case these objects are frame objects, the SDX dialog can connect objects automatically only if the corresponding interface objects exist in these frames and if those interface objects are specified in the mapping table (see chapter Map SDX Objects to Plant Simulation Objects below).

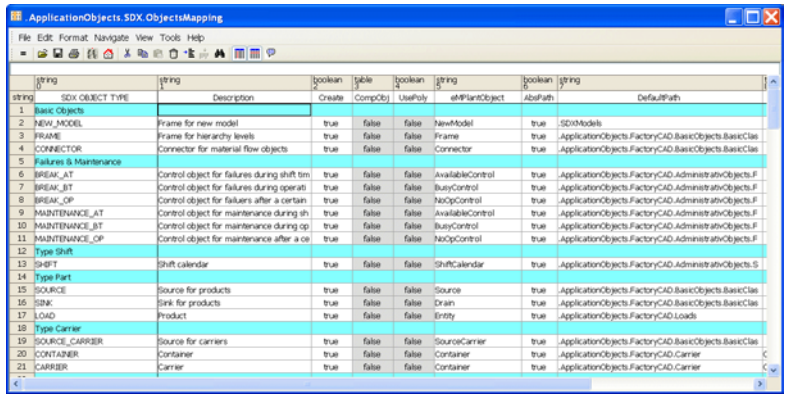
The SDX dialog can update the functionality listed above during the model update (see chapter Update Simulation Models above).

Map SDX Objects to Plant Simulation Objects

The mapping table specifies e.g.

- which objects should be used for the automatic model generation
- which Plant Simulation objects will be used for which objects in the SDX file for the automatic model generation
- which SDX attribute values should be written to which Plant Simulation attributes
- how to connect Plant Simulation objects (e.g. if interface objects should be used)

To open the mapping table, click on Edit Mapping Table on the Controls tab of the SDX dialog.



	string 0	string 1	boolean 2	table 3	boolean 4	string 5	boolean 6	string 7
string	SDX OBJECT TYPE	Description	Create	CompObj	UsePoly	eMPlantObject	AbsPath	DefaultPath
1	Basic Objects							
2	NEW_MODEL	Frame for new model	true	false	false	NewModel	true	SDXModels
3	FRAME	Frame for hierarchy levels	true	false	false	Frame	true	.ApplicationObjects.FactoryCAD.BasicObjects.BasicClas
4	CONNECTOR	Connector for material flow objects	true	false	false	Connector	true	.ApplicationObjects.FactoryCAD.BasicObjects.BasicClas
5	Failures & Maintenance							
6	BREAK_AT	Control object for failures during shift tim	true	false	false	AvailableControl	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.F
7	BREAK_BT	Control object for failures during operati	true	false	false	BusyControl	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.F
8	BREAK_OP	Control object for failuers after a certain	true	false	false	NoOpControl	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.F
9	MAINTENANCE_AT	Control object for maintenance during sh	true	false	false	AvailableControl	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.F
10	MAINTENANCE_BT	Control object for maintenance during op	true	false	false	BusyControl	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.F
11	MAINTENANCE_OP	Control object for maintenance after a ce	true	false	false	NoOpControl	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.F
12	Type Shift							
13	SHIFT	Shift calendar	true	false	false	ShiftCalendar	true	.ApplicationObjects.FactoryCAD.AdministrativObjects.S
14	Type Part							
15	SOURCE	Source for products	true	false	false	Source	true	.ApplicationObjects.FactoryCAD.BasicObjects.BasicClas
16	SINK	Sink for products	true	false	false	Drain	true	.ApplicationObjects.FactoryCAD.BasicObjects.BasicClas
17	LOAD	Product	true	false	false	Entity	true	.ApplicationObjects.FactoryCAD.Loads
18	Type Carrier							
19	SOURCE_CARRIER	Source for carriers	true	false	false	SourceCarrier	true	.ApplicationObjects.FactoryCAD.BasicObjects.BasicClas
20	CONTAINER	Container	true	false	false	Container	true	.ApplicationObjects.FactoryCAD.Carrier
21	CARRIER	Carrier	true	false	false	Container	true	.ApplicationObjects.FactoryCAD.Carrier

Picture 31:The mapping table

The mapping table contains the following columns:

SDX Object Type

This column shows the name of the SDX object type in the SDX file. In the SDX file the attribute name is INTERNAL_TYPE.

Description

The Description gives some explanation what this object type is used for.

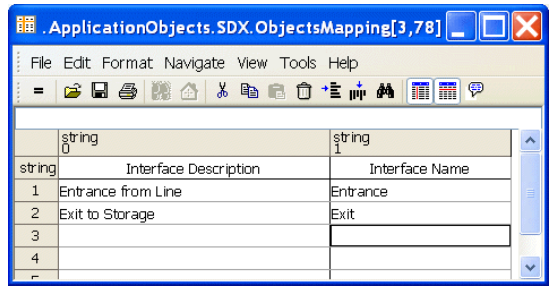
Create

Use the value true or false to define if the SDX dialog should insert objects of this type in the simulation model during model generation.

CompObj

Set this value to true, if the Plant Simulation object the SDX dialog should insert consists of a frame. Note, that the data type of this column is table. To open the table, right-click the field and select Open object from the context menu.

Use this table to specify the names of the interface objects of the frame object, in case that the SDX dialog should connect the object automatically with predecessor/successor objects. Note that this version supports only one entrance and one exit interface.



Picture 32:Interface table of the mapping table

UsePoly

Use this field to specify, if the SDX dialog should insert the Plant Simulation object as a block or draw a polyline curve (e.g. for segments of transport systems or paths). Valid values are **true** and **false**.

eMPlantObject

This field should contain the Plant Simulation name of the object the SDX dialog should insert.

AbsPath

Use this field to specify if the field **DefaultPath** (see below) contains the absolute path to the Plant Simulation object the SDX dialog should insert (**true**) or if the Plant Simulation object is located in the same folder where the control object for this Plant Simulation object is located (**false**).

The benefit of using an absolute path is that you could use e.g. the same Plant Simulation object for the same station type, segment type etc. for different types of transport systems. In this case, you need the station/segment object just once in the class library. If the SDX dialog should use different station/segment objects for different types of transport systems, set the value of **AbsPath** to **false** and the SDX dialog will look for the station/segment object under the specified name in the **DefaultPath** folder specified for the control object of the transport system.

DefaultPath

This field specifies the folder where the Plant Simulation object is located.

Attributes

The mapping table does not only allow a mapping of object types but also a mapping of SDX attributes to attributes of Plant Simulation objects. To change the mapping, right-click the **Attributes** field and choose **Open object** from the context menu. This table contains the following columns:

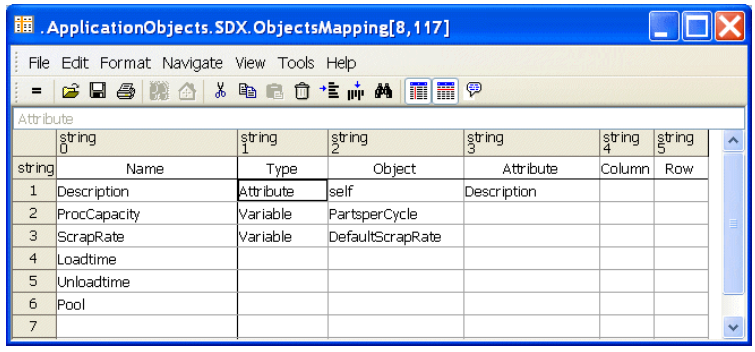
1. The column **Name** specifies the name of the attribute in the Creation Table

Note: To view the attributes in the Creation Table of a simulation model, double click the **Data** object in the top section of the simulation model and click on **Open Creation Table**. This table has a column **Attributes** (which is of the data type table itself). Right-click a field in this column which contains a table and click on Open Object to open this sub table. This sub table contains all the attributes the SDX dialog uses for the model generation

2. The column **Type** specifies what the SDX dialog should do with the attribute value:

- For the type **Attribute**, the SDX dialog copies the value of this attribute directly to an attribute of an Plant Simulation object.
- For the type **Table**, the SDX dialog copies the value of this attribute to an Plant Simulation table.
- For the type **Variable**, the SDX dialog copies the value of this attribute to an Plant Simulation variable object.
- For the type **Method**, the SDX dialog calls the method specified and passes the attribute value as a parameter for this method.

3. The column Object specifies where the SDX dialog should copy the value. Depending on the Type, this object should be an Plant Simulation table, variable, method or any object which has the required attribute.
4. The column Attribute should contain the name of the attribute of the Plant Simulation object. This column is only relevant if the Type is Attribute.
5. The columns Column and Row specify in which column and row of an Plant Simulation table the SDX dialog should copy the attribute value. This column is only relevant if the Type is Table. The column should contain the index of the column/row.



string	string 0 Name	string 1 Type	string 2 Object	string 3 Attribute	string 4 Column	string 5 Row
1	Description	Attribute	self	Description		
2	ProcCapacity	Variable	PartsperCycle			
3	ScrapRate	Variable	DefaultScrapRate			
4	Loadtime					
5	Unloadtime					
6	Pool					
7						

Picture 33:Attribute table of the mapping table

ImporterObject

In case the Plant Simulation object the SDX dialog should use for model generation consists of a frame which contains other Plant Simulation objects (like most application objects in case you work with application object libraries), the column ImporterObject should contain the name of the object inside this application object which allocates required importers like workers/operators etc.

Note: Usually application objects contain a Plant Simulation SingleProc object which allocates workers/operators for this application object. Then this SingleProc is the ImporterObject for this application object.

Customizing the Model Generation Process

The Advanced Tab of the SDX Dialog

You can execute additional SimTalk code

1. directly after the SDX dialog filled the Creation Table, just before model generation or
2. directly after model generation (see the chapter Steps of a Model Generation Process above).

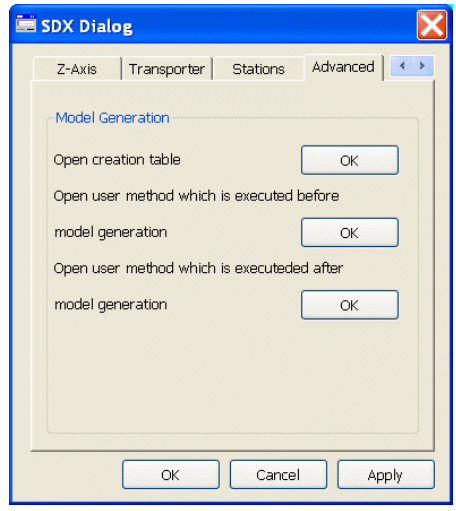
To open the Creation Table, click on Open Creation Table.

Note: The Creation Table contains data only after step 3 of the model generation process (see the chapter Steps of a Model Generation Process above) which is executed when you click on Generate Simulation Model on the Controls tab.

To execute SimTalk code just before the model generation, click on Open user method which is executed before model generation. Use this method e.g. to enter additional objects in the Creation Table which you would like the SDX dialog to create in the simulation model.

To execute SimTalk code directly after model generation, click on Open user method which is executed after model generation. Use this method e.g. to set additional parameters, import schedules or production plans from databases, open custom dialogs etc.

Note: Since these methods are located inside the SDX folder, the method text in these methods will get lost when you upgrade the version of the SDX.obj file to a higher patch version. If you would like to implement methods which will not be affected by version upgrades, create your own customization file, see below.



Picture 34:The Advanced tab of the SDX dialog

Working with a Customization File

The main idea of working with a customization file is that it allows you to customize

1. the model generation process, e.g. creating additional objects, implementing additional logic
2. the objects to be created by replacing the mapping table with a custom mapping table
3. the model by executing additional SimTalk code, e.g. to load scheduling information or production plans from databases, open custom dialogs etc.

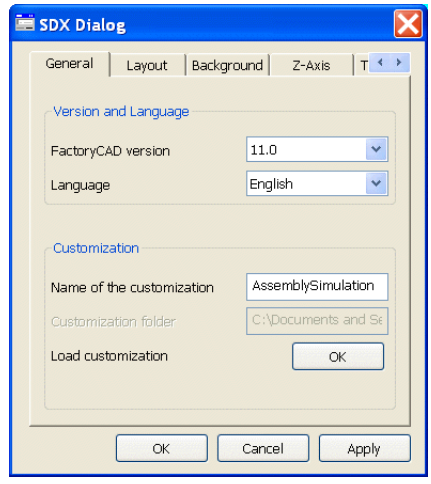
independently from the SDX.obj file. This enables you to upgrade the SDX.obj file and your customization file independently from each other and without overwriting information in the other file.

Requirements of Working with a Customization File

Customization Name

To enable working with customizations, every customization needs a name, e.g. "AssemblySimulation". The preferences and the internal logic will refer to this name (e.g. looking for objects under a folder of this name in the class library).

Enter the name of the customization in the SDX dialog on the General tab in the field Name of the Customization.



Picture 35:The General tab of the SDX dialog

Customization Object

All the custom logic, objects, dialogs etc. should be part of a separate Plant Simulation.obj file. The name of the file should be the customization name, e.g. “AssemblySimulation.obj“.

This object should be located in the folder

C:\Documents and Settings\All Users\Application Data\Tecnomatix\Plant Simulation

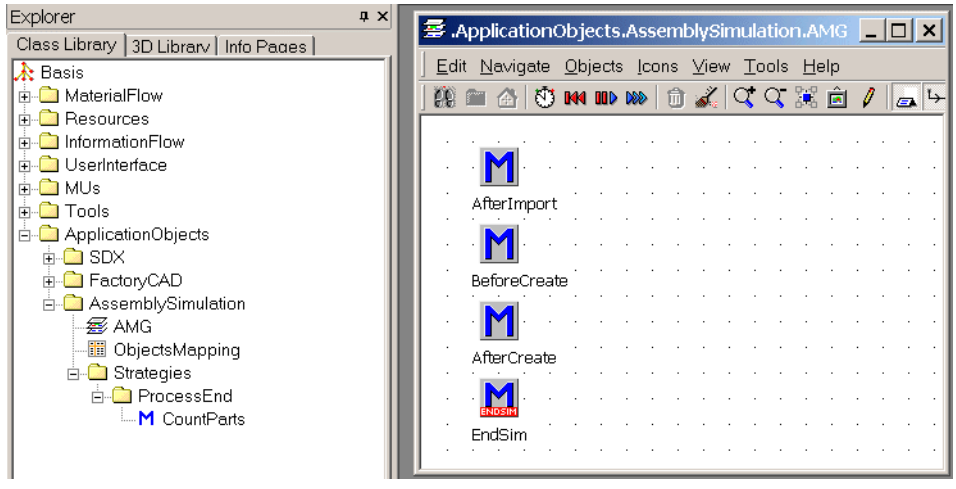
or the corresponding folder of your operating system.

Customization Folder

The Plant Simulation objects from the customization should be located in a separate folder with the name of the customization in the Plant Simulation class library. This folder should be located under the “.ApplicationObjects” folder, e.g. “.ApplicationObjects.AssemblySimulation“.

Reserved Method Names

If there is a frame “AMG“ (for “Automatic Model Generation“) under the customization folder in the Plant Simulation class library which contains methods of certain names, the SDX dialog will automatically execute these methods in the corresponding phase of the model generation.



Picture 36:The folder for customizations and the AMG frame

These method names are:

1. AfterImport

   The SDX dialog will execute this method before it fills the Creation Table (between phase 2 and 3, see the chapter Steps of a Model Generation Process above). This enables you to modify the data in the SDX data tables which the SDX dialog uses to derive the content of the Creation Table.

2. BeforeCreate

   The SDX dialog will execute this method after it fills the Creation Table but before it creates the simulation model (between phase 3 and 4 of the model generation process). This enables you to modify the data in the Creation Table which the SDX dialog uses to build up the simulation model.

3. AfterCreate

   The SDX dialog will execute this method after it created the simulation model (after phase 4 of the model generation process). This enables you to modify the simulation model before the start of the simulation.

4. EndSim

   The SDX dialog will execute this method after the end of each simulation run.

Note: In case you execute multiple simulation runs using different random numbers ("observations"), the SDX dialog executes the EndSim method after the last run only.

Mapping Table

If there is a table ObjectsMapping under the customization folder in the class library, the SDX dialog will automatically replace the default mapping table (located under .ApplicationObjects.SDX) with this custom mapping table when loading the customization.

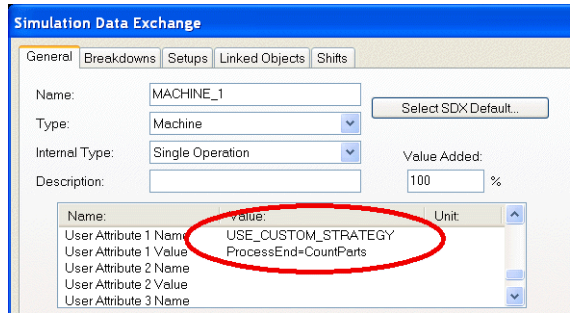
To create a table ObjectsMapping for a customization, derive this table from the ObjectsMappingClass located under .ApplicationObjects.SDX in the class library.

Strategies

Plant Simulation objects from the FactoryCAD application objects library contain multiple user methods, where you can enter additional SimTalk code (e.g. to realize special strategies, count parts, collect results etc.). Using a customization, you can automatically replace those user methods with methods from your customization.

To use this functionality, there must be a folder "Strategies" under your customization folder. Under this folder, there should be a folder with the name of the user method (e.g. "ProcessEnd"). This folder should contain all the possible custom methods for this user method, e.g. you might write a method which counts the number of parts - so you name this method "CountParts" (see the picture The folder for customizations and the AMG frame above).

Now if you define a custom attribute with the name "USE_CUSTOM_STRATEGY" for a SDX object and the value of this attribute is <name of the user method>=<name of the custom method>, e.g. ProcessEnd=CountParts, then the SDX dialog automatically replaces the ProcessEnd method for this object with the custom method during model generation.



Picture 37:The USE_CUSTOM_STRATEGY attribute of SDX objects

Note: If there is a new version of the SDX data and you added or removed the USE_CUSTOM_STRATEGY attribute to/from the SDX object, then the SDX dialog will add/remove this custom strategy for this object.

Note: If you developed a new version of your custom method and you would like to update existing Plant Simulation models, just load the customization object with "Update class library". The user methods replaced by the SDX dialog are actually instances of the methods from the customization folder, so they will be updated automatically.

Loading a Customization

When you entered a customization name on the General tab of the SDX dialog, then the SDX dialog will load the customization as soon as you start the SDX.spp file.

If you would like to load the customization manually (e.g. to update the version of the customization in your simulation model manually), click on the Load customization button on the General tab of the SDX dialog.

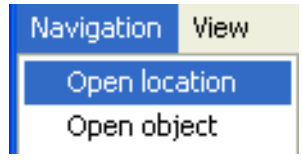
The Plant Simulation FactoryCAD Objects

General Menus

All Plant Simulation FactoryCAD object dialogs share two menus, the Navigation menu including Open location and the Open object selection, and the View menu which includes at least the Refresh option, depending on the object the options Content, Open segment and/or Control object can be added.

The Navigation Menu

Contains normally two options: Open location and Open object. Use it to navigate through your model.



Picture 1:The navigation menu

Open Location

This option opens the location of the object, most of the time it is your model.

Open Object

Opens the Plant Simulation basic object underneath the Plant Simulation FactoryCAD object, normally a frame or a transport object like a line or track.

The View Menu

Can have different entries depending on the object. At least the Refresh option is given, on material handling devices additional Content is available.

Refresh

Reloads the dialog, e.g. with new status information.

Content

This menu can be added if the Plant Simulation FactoryCAD object is a transport device like a conveyor or a monorail, it shows all objects and their position on the transport device.

Open Segment

If the object is a station of a pivot, you can open the dialog of the pivot segment with this menu.

Control Object

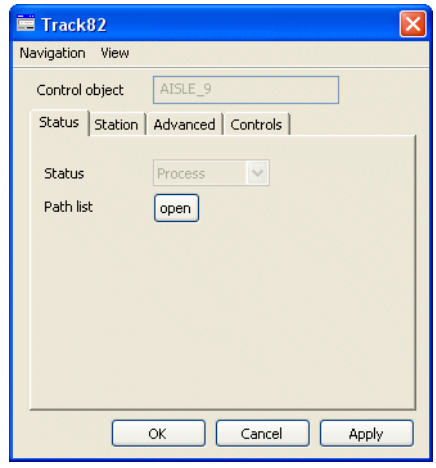
Most Plant Simulation FactoryCAD objects are linked with other objects, e.g. the segments of a conveyor system. This link is done with the control object, which links all corresponding object together.

General Layout of the Dialogs

All dialogs share some common features:

- The title of the dialog is the name of the SDX object. If there is no SDX name for this object, then the title is the name of the Plant Simulation object.
- If it got one, a field stating the name of the control object is the next field.

- They show a tab area with different tabs, here a few examples of common tabs:
  - Status
  - Station
  - Data
  - Advanced
  - Controls



The Tab Status

The status of an object includes if it is failed or in working order, if it is paused, unplanned or available, how many parts are on it and how many times it has worked.

The Tab Station

Contains all information about the station, like the type of the station, operation time or plan.

The Tab Data

The tab Data is only found at buffers, machines and the like. It contains the essential data of these objects.

The Tab Advanced

Here some more advanced features are implemented. Those features are not available in FactoryCAD or the Route Editor of FactoryCAD. The features available depend entirely on the type of object.

The Tab Controls

Normally the controls consist of two decision methods, one entry and one exiting method. The default behavior for the entry control is to take the first part which comes, the default behavior of the exit control is to distribute the parts per random to the successors.

Transport Objects

These objects are normally length-oriented, which means that their capacity is restricted by their physical dimensions. Point oriented objects are limited by capacity, e.g. a drilling machine can only process one part a time.

The Floorconveyor Objects

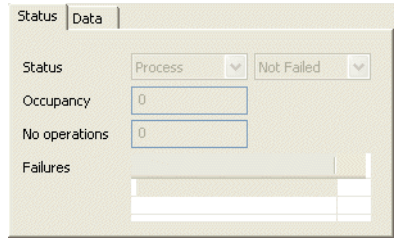
Floorconveyors are the most common transport system. It is possible to get a part on a conveyor without a load station.

The Control Object

Controls a sequence of segments of a floorconveyor, their speed, acceleration, failures and shifts. A sequence of a floorconveyor consists of several segments, stations or basic segments.

Tab Status

Contains data as the status of the objects and failures.



Picture 2:A not paused or failed object without a failure generator

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Number of Operations

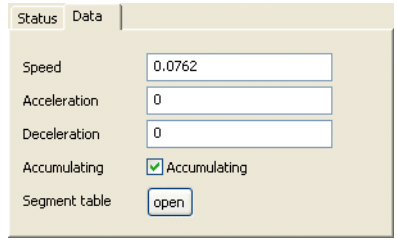
This field shows the quantity of parts which entered the segment.

Failures

A link to the failure generators of the object. You find detailes about failure generators in the chapter Administrative Objects - Failure Definition.

Tab Data

Contains all data specific for this type of object.



Picture 3:A accumulating conveyor without acceleration

Speed

The maximum speed of the conveyor.

Acceleration and Deceleration

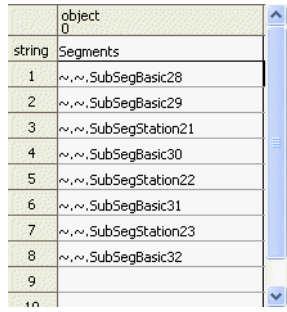
Defines if acceleration and deceleration is used. A value of zero in acceleration means it is not used.

Accumulating

If the conveyor is rigid, uncheck the checkbox, otherwise the conveyor accumulates packages on itself.

Segment Table

A table with all segment objects in it for random access.

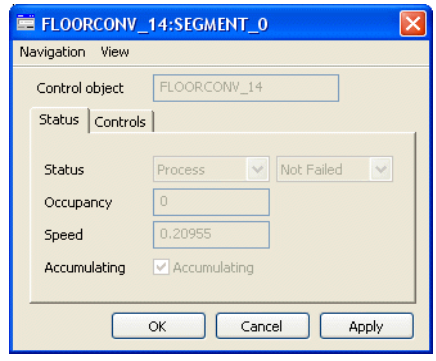


	object 0
string	Segments
1	~.~.SubSegBasic28
2	~.~.SubSegBasic29
3	~.~.SubSegStation21
4	~.~.SubSegBasic30
5	~.~.SubSegStation22
6	~.~.SubSegBasic31
7	~.~.SubSegStation23
8	~.~.SubSegBasic32
9	

Picture 4:A segment table of a floorconveyor control

The Basic Segment Object

A segment without a station, containing only the basic data: length and successors. The dialog contains a link to the control object, and two tabs: Status and Controls.



Picture 5:The dialog of a basic segment.

Special Menus in the Floorconveyor Basic Segment

Additional to the normal menus the basic segment has the menu items Contentlist and Control object.

Menu Item Contentlist

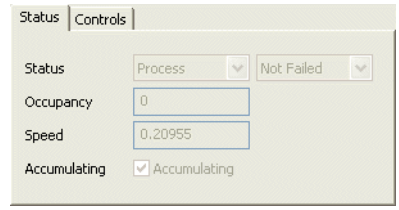
Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.



Picture 6:The status tab of a basic segment

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Speed

The actual speed of the conveyor.

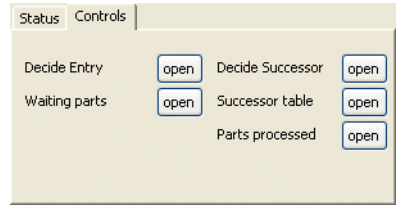
Accumulating

If the conveyor is rigid, the checkbox is unchecked, otherwise the conveyor accumulates packages on itself.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

Tab Controls

The two methods concerning the entrance and the exit of the segment are reachable on this tab.



Picture 7:The controls tab of a basic segment

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the object.

Successor Table

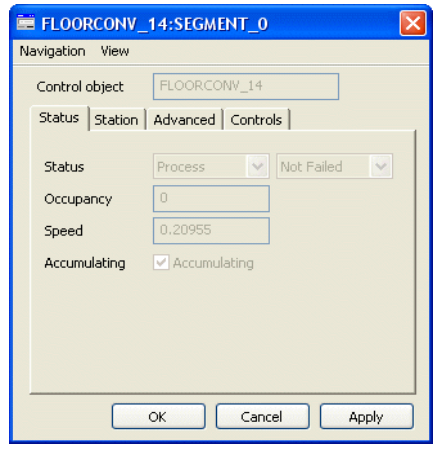
Opens a table with all the possible successor objects.

Parts processed

Opens the table which contains all parts waiting for transport and their designated target.

The Station Segment Object and the Station

The station consists of two different objects, the station segment, which is the transport device of a station, and the station itself, which is the working device of the station. They share the dialog and the control object. An additional tab for the processing data is given.



Picture 8:Dialog of a conveyor station

Special Menus in the Floorconveyor Station Segment

Additional to the menus of a basic floorconveyor segment the station segment has the menu item Open station in the Navigation menu.

Menu Item Open Station

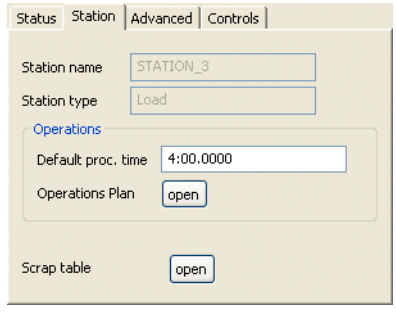
Opens the frame of the Plant Simulation object which represents the station part of the segment.

Tab Status

This tab shows the status of the segment (paused or failed), occupancy and speed.

Tab Station

Contains the values concerning the station part of the segment.



Picture 9:The tab station of a regular monorail station.

Station Name

The name of the station in the SDX data.

Station Type

The type of the station in the SDX data. Load, Unload, ProcessStop and ProcessRun are the most commonly used.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

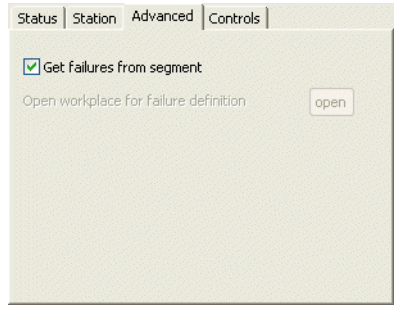
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 10:The advanced tab of a default station

Get Failures from Segment

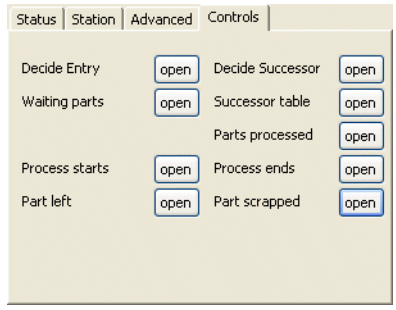
Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

There are more controls on a station than on a basic segment. In addition to the entrance and exit controls, there are methods which are called when specific events occur, like the beginning of processing the part.



Picture 11:The controls tab of a station segment

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the object.

Successor Table

Opens a table with all the possible successor objects.

Parts Processed

Opens the table which contains all parts waiting for transport and their designated target.

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

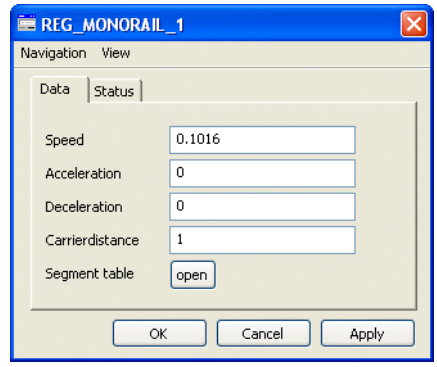
This method is called when a part is scrapped on the station.

The Regular Monorail Objects

Monorail conveyors are single-track conveyors that move loads hanging from an overhead track. The track itself is a beam, with trollies that travel along the beam as they are pulled by a chain.

The Control Object

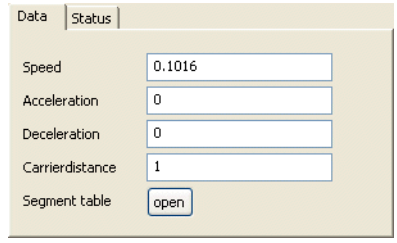
Synchronizes all segments of the regular monorail, controls the failures, the speed and the acceleration.



Picture 12:The control object of a regular monorail.

Tab Data

Contains all data specific for this type of object.



Picture 13:The data of a regular monorail.

Speed

The maximum speed of the monorail.

Acceleration and Deceleration

Defines if acceleration and deceleration is used. A value of zero in acceleration means it is not used.

Carrierdistance

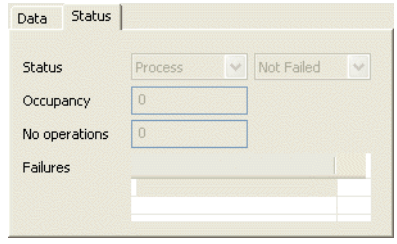
Defines the minimal distance between two carriers.

Segment Table

A table with all segment objects in it for random access.

Tab Status

Contains general data at the status of the objects and failures.



Picture 14:A not paused or failed object without a failure generator

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Number of Operations

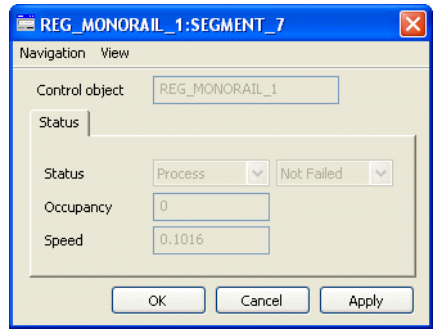
Number of parts that were on the segments.

Failures

A link to the failure generators of the object.

The Basic Segment Object

A bare segment, just a monorail. The dialog just contains links to the control object and the Plant Simulation object, and the general status of the regular monorail.



Picture 15:The dialog of a basic regular monorail segment

Special Menus in the Regular Monorail Basic Segment

Additional to the normal menus the basic segment has the menu items Contentlist and Control object.

Menu Item Contentlist

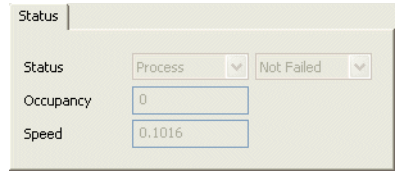
Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.



Picture 16:The status tab of a basic segment

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

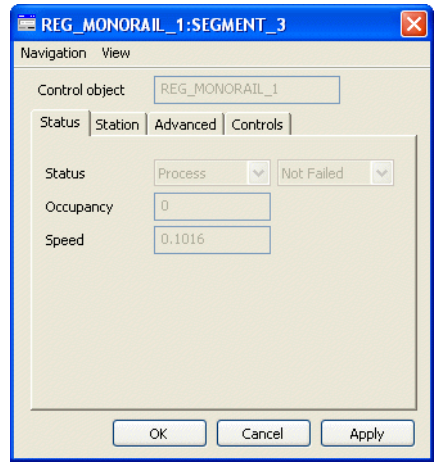
Speed

The actual speed of the chain.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

The Station Segment Object and the Station

The station consists of two different objects, the station segment, which is the transport device of a station, and the station itself, which is the working device of the station. They share the dialog and the control object. An additional tab for the processing data is given.



Picture 17:Dialog of a regular monorail station

Special Menus in the Regular Monorail Station Segment

Additional to the menus of a basic regular monorail segment the station segment has the menu item Open station in the Navigation menu.

Menu Item Open Station

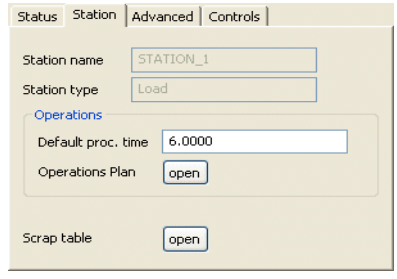
Opens the frame of the Plant Simulation object which represents the station part of the segment.

Tab Status

This tab shows the status of the segment (paused or failed), occupancy and speed.

Tab Station

Contains the values concerning the station part of the segment.



Picture 18:The tab station of a conveyor station.

Station Name

The name of the station, as given in FactoryCAD.

Station Type

The type of the station, as given in FactoryCAD. Load, Unload, ProcessStop and ProcessRun are the most commonly used.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

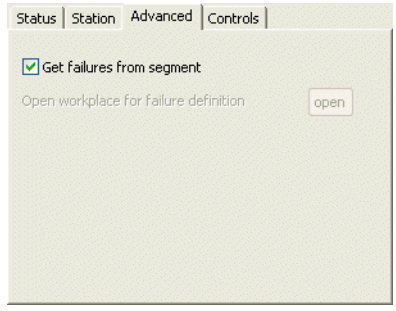
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 19:The advanced tab of a default station

Get Failures from Segment

Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

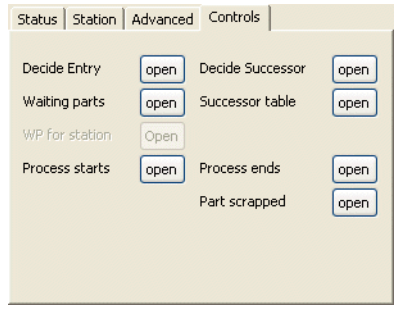
Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

There are more controls on a station than on a basic segment. In addition to the entrance and exit controls, there are methods which are called when specific events occur, like the beginning of processing the part.

Some Controls are only available for specific stations, e.g. the Decide Entry control is only for Load stations, the WP for station is only available for Assembly stations.



Picture 20:The tab Controls of a regular monorail station

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance (only available for Load stations).

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the station (only available for Load stations).

Successor Table

Opens a table with all the possible successor objects.

WP for Stations

Opens the a table with the parts waiting to be assembled at the station (attachment parts, only available for Assembly stations).

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

This method is called when a part is scrapped on the station.

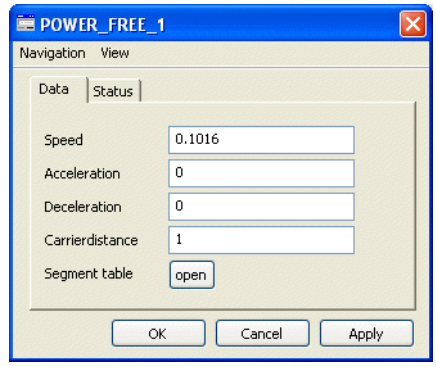
The Power and Free Objects

Power-and-free conveyors use two tracks: the power track uses a powered chain to move the load, which slides freely along the free track. On the powered chain track, a hook mechanism called a dog contacts a trolley on the free track to pull the load along

Note: The powered track must form a complete loop.

The Control Object

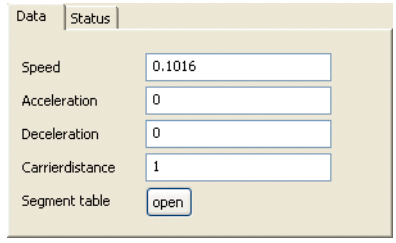
Synchronizes all segments of the power and free monorail, controls the failures, the speed and the acceleration.



Picture 21:The control object of a power and free system

Tab Data

Contains all data specific for this type of object.



Picture 22:The data of a regular monorail.

Speed

The maximum speed of the monorail.

Acceleration and Deceleration

Defines if acceleration and deceleration is used. A value of zero in acceleration means it is not used.

Carrierdistance

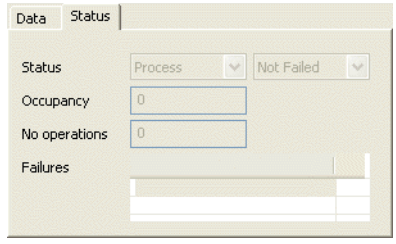
Defines the minimal distance between two carriers.

Segment Table

A table with all segment objects in it for random access.

Tab Status

Contains general data at the status of the objects and failures.



Picture 23:A not paused or failed object without a failure generator

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Number of Operations

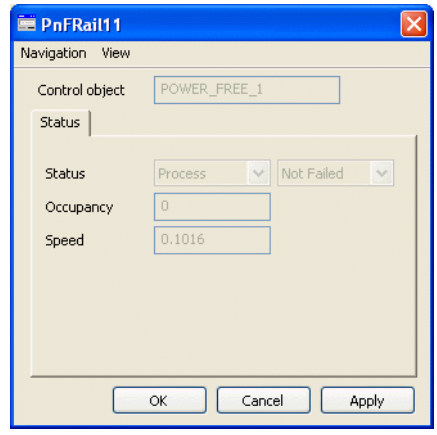
Number of parts that were on the segments.

Failures

A link to the failure generators of the object.

The Basic Segment Object

A bare segment, just a monorail. The dialog just contains links to the control object and the Plant Simulation object, and the general status of the power and free system.



Picture 24:The dialog of a basic regular monorail segment

Special Menus in the Power and Free Basic Segment

In addition to the normal menus the basic segment has the menu items Contentlist and Control object.

Menu Item Contentlist

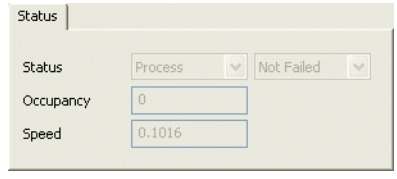
Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.



Picture 25:The status tab of a basic segment of a power and free system

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

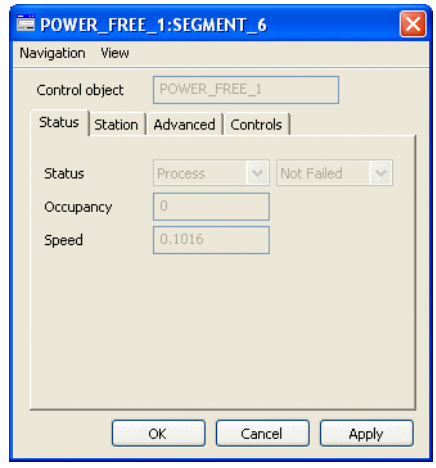
Speed

The actual speed of the chain.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

The Station Segment Object and the Station

The station consists of two different objects, the station segment, which is the transport device of a station, and the station itself, which is the working device of the station. They share the dialog and the control object. An additional tab for the processing data is given.



Picture 26:Dialog of a regular monorail station

Special Menus in the Power and Free Station Segment

In addition to the menus of a basic power and free segment the station segment has the menu item Open station in the Navigation menu.

Menu Item Open Station

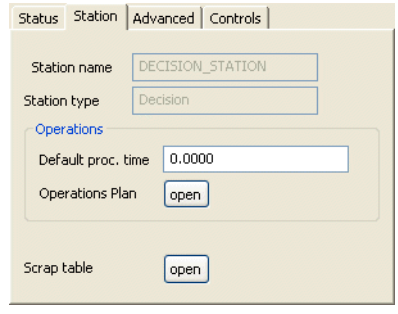
Opens the frame of the Plant Simulation object which represents the station part of the segment.

Tab Status

This tab shows the status of the segment (paused or failed), occupancy and speed.

Tab Station

Contains the values concerning the station part of the segment.



Picture 27:The tab station of a conveyor station.

Station Name

The name of the station, as given in FactoryCAD.

Station Type

The type of the station, as given in FactoryCAD. Load, Unload, ProcessStop and ProcessRun are the most commonly used.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

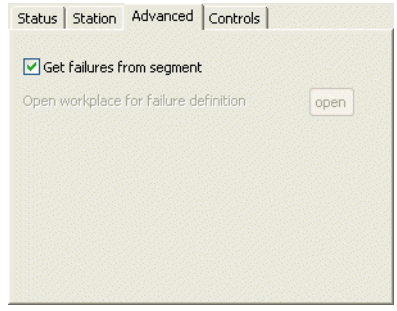
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 28:The advanced tab of a default station

Get Failures from Segment

Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

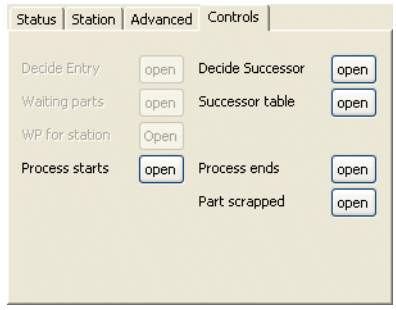
Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

There are more controls on a station than on a basic segment. In addition to the entrance and exit controls, there are methods which are called when specific events occur, like the beginning of processing the part.

Some Controls are only available for specific stations, e.g. the Decide Entry control is only for Load stations, the WP for station is only available for Assembly stations.



Picture 29:The tab Controls of a power and free station

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance (only available for Load stations).

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the station (only available for Load stations).

Successor Table

Opens a table with all the possible successor objects.

WP for Stations

Opens the a table with parts waiting to be assembled at the station (attachment parts, only available for Assembly stations).

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

This method is called when a part is scrapped on the station.

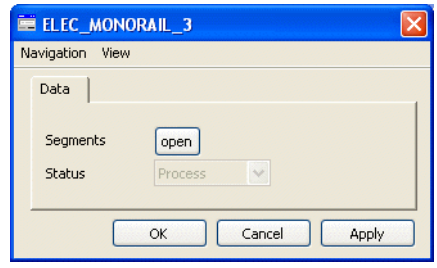
The Electrified Monorail Objects

Electrified monorail system (EMS) rails are single-track rails that have electrical power lines along the track. A pusher motor draws power from the track and moves the load along the track with a trolley.

Electrified monorail systems are the most flexible types of transport devices, because they do not require a closed loop for the track, and they can have junctions that transfer loads from one track to another to create complex networks.

The Control Object

It controls just the shift for its segments. The dialog contains the segments list and the status of the monorail.



Picture 30:The control object of a electrified monorail.

Tab Data

Contains all data specific for this type of object.

Segment Table

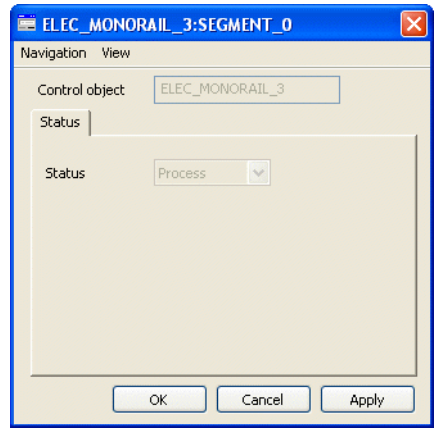
A table with all segment objects in it for random access.

Status

This tab shows the status of the transport system: process, paused, failed or unplanned (meaning there's no shift).

The Basic Segment Object

A bare segment, just a monorail. The dialog just contains links to the control object and the Plant Simulation object, and the general status of the electrified monorail.



Picture 31:The dialog of a basic electrified monorail segment

Special Menus in the Electrified Monorail Basic Segment

In addition to the normal menus the basic segment has the menu items Contentlist and Control object.

Menu Item Contentlist

Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

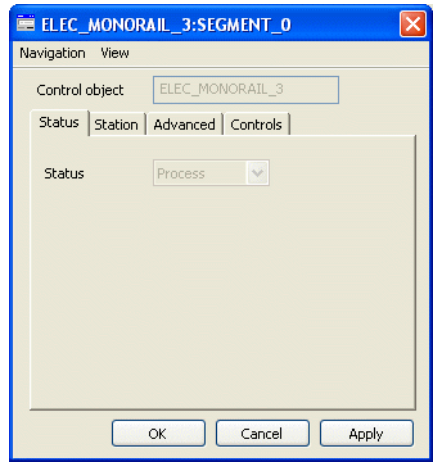
This tab shows the status of the segment: Paused or not.

Status

This field shows the status of the segment: process, paused, failed or unplanned (meaning there's no shift).

The Station Segment Object and the Station

The station consists of two different objects, the station segment, which is the transport device of a station, and the station itself, which is the working device of the station. They share the dialog and the control object. An additional tab for the processing data is given.



Picture 32:Dialog of a electrified monorail station

Special Menus in the Electrified Monorail Station Segment

In addition to the menus of a basic electrified monorail segment the station segment has the menu item Open station in the Navigation menu.

Menu Item Open Station

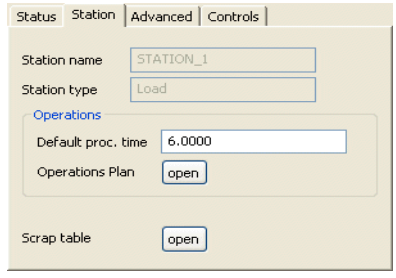
Opens the frame of the Plant Simulation object which represents the station part of the segment.

Tab Status

This tab shows the status of the segment (paused or failed), occupancy and speed.

Tab Station

Contains the values concerning the station part of the segment.



Picture 33:The tab station of a electrified monorail station.

Station Name

The name of the station, as given in FactoryCAD.

Station Type

The type of the station, as given in FactoryCAD. Load, Unload, ProcessStop and ProcessRun are the most commonly

used.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

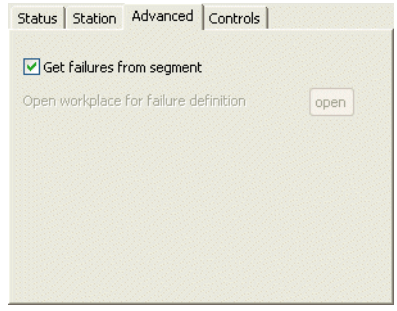
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 34:The advanced tab of a default station

Get Failures from Segment

Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

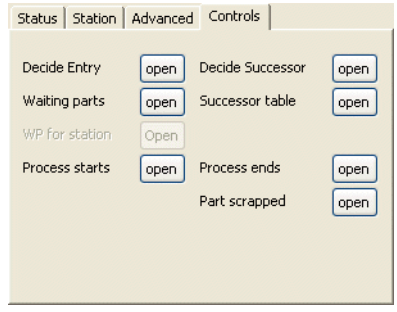
Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

There are more controls on a station than on a basic segment. In addition to the entrance and exit controls, there are methods which are called when specific events occur, like the beginning of processing the part.

Some Controls are only available for specific stations, e.g. the Decide Entry control is only for Load stations, the WP for station is only available for Assembly stations.



Picture 35:The tab Controls of a electrified monorail station

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance (only available for Load stations).

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the station (only available for Load stations).

Successor Table

Opens a table with all the possible successor objects.

WP for Stations

Opens the a table with parts waiting to be assembled at the station (attachment parts, only available for Assembly stations).

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

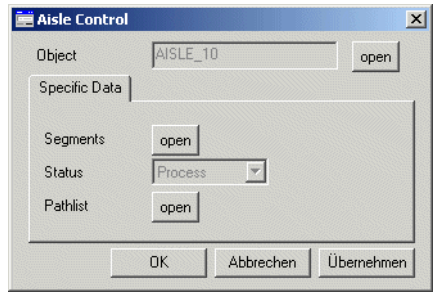
This method is called when a part is scrapped on the station.

The Aisle Objects

An aisle is an open passage between rooms, machinery, or stored materials, and is used by people and material handling machinery for moving people and loads.

The Control Object

It controls just the shift for its segments. The dialog contains the segments list and the status of the aisles.



Picture 36:The control object of a aisle.

Tab Data

Contains all data specific for this type of object.

Segment Table

A table with all segment objects in it for random access.

Status

This field shows the status of the aisle.

The Basic Segment Object

A bare segment, just a track. The dialog just contains links to the control object and the Plant Simulation object, and the general status of the aisle.

Picture 37:The dialog of a basic aisle track

Special Menus in the Aisle Basic Segment

In addition to the normal menus the basic segment has the menu items Contentlist and Control object.

Menu Item Contentlist

Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment and the PathList

Status

This field shows the status of the segment: process or failed.

The Station Segment Object and the Station

The station consists of two different objects, the station segment, which is the transport device of a station, and the station itself, which is the working device of the station. They share the dialog and the control object. An additional tab for the processing data is given.

Picture 38:Dialog of a aisle station

Special Menus in the Aisle Station Segment

In addition to the menus of a basic aisle segment the station segment has the menu item Open station in the Navigation menu.

Menu Item Open Station

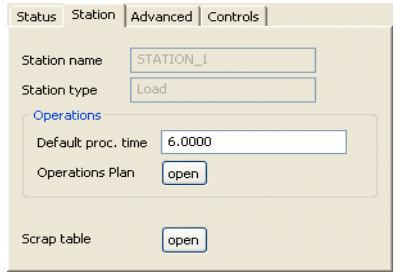
Opens the frame of the Plant Simulation object which represents the station part of the segment.

Tab Status

This tab shows the status of the segment (paused or failed), occupancy and speed.

Tab Station

Contains the values concerning the station part of the segment.



Picture 39:The tab station of a aisle station.

Station Name

The name of the station, as given in FactoryCAD.

Station Type

The type of the station, as given in FactoryCAD. Load, Unload, ProcessStop and ProcessRun are the most commonly used.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

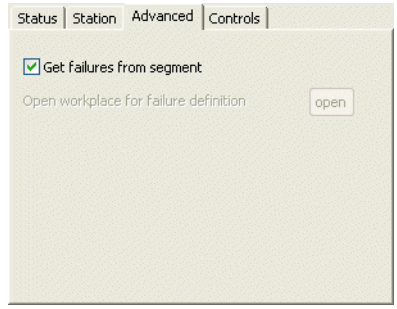
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 40:The advanced tab of a default station

Get Failures from Segment

Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

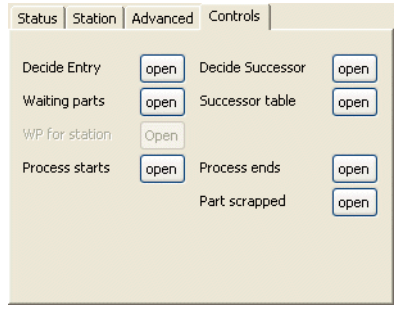
Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

There are more controls on a station than on a basic segment. In addition to the entrance and exit controls, there are methods which are called when specific events occur, like the beginning of processing the part.

Some Controls are only available for specific stations, e.g. the Decide Entry control is only for Load stations, the WP for station is only available for Assembly stations.



Picture 41:The tab Controls of a aisle station

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance (only available for Load stations).

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the station (only available for Load stations).

Successor Table

Opens a table with all the possible successor objects.

WP for Stations

Opens the a table with parts waiting to be assembled at the station (attachment parts, only available for Assembly stations).

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

This method is called when a part is scrapped on the station.

The Pivot Objects

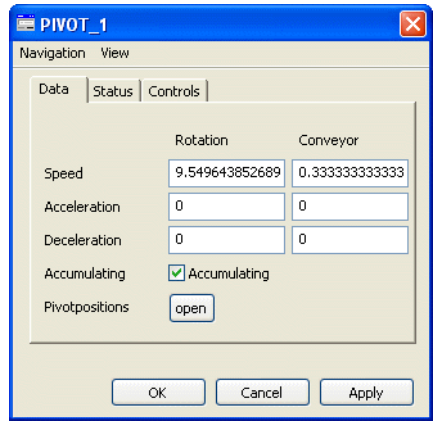
There are two different sets of pivot objects, one for the floorconveyor and one for the electrified monorail. A set of pivot objects contains the control object, the pivot object and the stations. To use the pivot, at least a control object, the pivot and two stations are needed.

The Pivot as a Part of a Floorconveyor

The pivoting segment is an active segment, with its own transport speed and rotation speed.

The Control Object

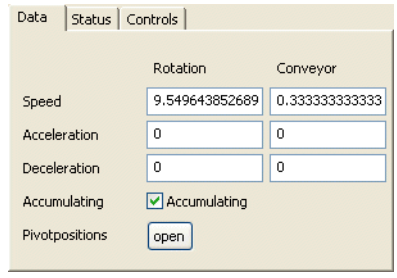
The control object of the pivot contains all data needed for the physical attributes of a pivot like length etc.



Picture 42:The dialog of a conveyor pivot control object.

Tab Data

On this tab all information concerning the pivot is gathered.



Picture 43:The data tab of a conveyor pivot control object

Rotation Speed, Acceleration and Deceleration

The column contains the data for the rotation behavior. Speed is the maximum speed, acceleration and deceleration define if the rotation is a accelerated movement or not. An acceleration value of zero means no acceleration.

Conveyor Speed, Acceleration and Deceleration

This values corresponds to the same values of the conveyor control object.

Accumulating

This value corresponds to the same values of the conveyor control object.

Pivotpositons

The button opens a table where the positions of the different stations are defined.

	string 0	integer 1	object 2	object 3
string	PivotPosition	IconAngle	InStations	OutStations
1	54	332	~.~.General3	~.~.General2
2	57	242	~.~.General5	~.~.General4
3				

Picture 44:The position table of a pivot

Each line in this table defines one position of the pivot segment

PivotPosition

This is the name of this position. It is purely artificial for recognizing this position.

IconAngle

The integer value defines the angle of the pivot segment in this position. Range is between 0 and 360 degrees.

InStations

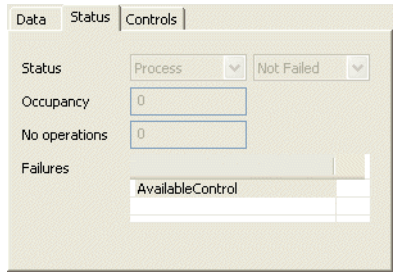
Gives an object link to the entry station on this pivot position.

OutStations

Gives an object link to the exit station on this pivot station.

Tab Status

On this tab all information concerning the state of the pivot is gathered.



Picture 45:The status tab of a conveyor pivot control object

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

How many parts are on the pivot.

Number of Operations

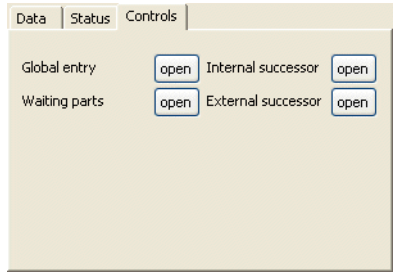
Number of parts that were on the pivot.

Failures

A link to the failure generators of the object.

Tab Controls

The methods and tables concerning the material flow are available on this tab.



Picture 46:The controls of a conveyor pivot control object

Global Entry

This method decides which part enters the pivot, when there is more than one part at the entrance stations.

Waiting Parts

Opens the table with the parts waiting to enter the stations.

Internal Successor

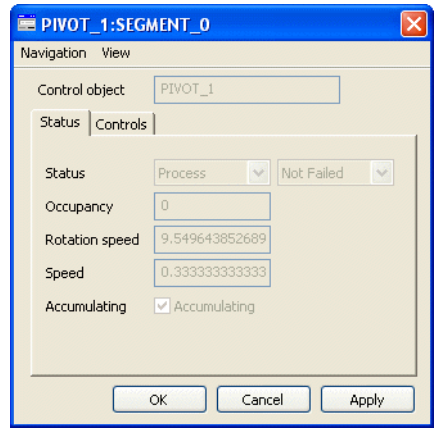
Opens a method which decides based on the part and on the station the part is located to which station the pivot will turn.

External Successor

The method which opens with this button assumes that the pivot has reached its end position, and is now trying to unload the part to a successor.

The Pivot Segment Object

In the case of a conveyor pivot, this object is a conveyor segment which will be rotated by the system.



Picture 47:The dialog of the pivot segment

Special Menus in the Floorconveyor Pivot Segment

In addition to the normal menus the pivot segment has the menu items Contentlist and Control object.

Menu Item Contentlist

Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

How many parts are on the pivot.

Rotation Speed

The actual rotation speed of the pivot.

Speed

The actual speed of the conveyor.

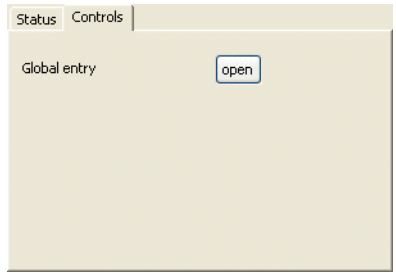
Accumulating

If the conveyor is rigid, the checkbox is unchecked, otherwise the conveyor accumulates packages on itself.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

Tab Controls

On the pivot segment only the method for the global entry is linked.



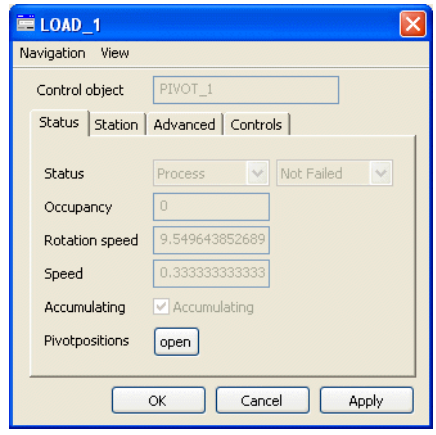
Picture 48:The status tab of a pivot segment

Decide Entry

This method decides which part enters the pivot, when there is more than one part at the entrance stations

The station

Stations are the entry and exit points of the pivot and are used for the control strategies.



Picture 49:The dialog of a pivot station

Special Menus in the Floorconveyor Pivot Station

In addition to the normal menus the pivot segment has the menu items Open Segment and Control object.

Menu Item Open Segment

Opens the dialog of the basic object of the pivot segment.

Menu Item Control Object

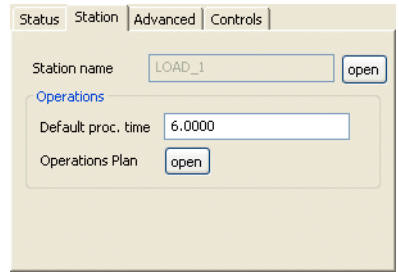
Opens the dialog of the control object of the segment.

Tab Status

This tab is the same as the one on the pivot segment.

Tab Station

On this tab all information about the station is assembled.



Picture 50:The station tab of a pivot station

Station Name

The name of the station, as given in FactoryCAD.

Default Proc Time

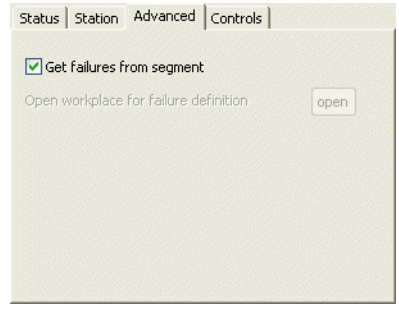
The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 51:The advanced tab of a pivot station

Get Failures from Segment

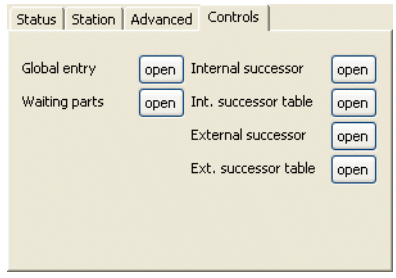
Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

The methods and tables concerning the material flow are available on this tab.



Picture 52:The controls of a pivot station

Global Decide Entry

This method decides which part enters the lift, when there is more than one part at the entrance stations.

Waiting Parts

Opens the table with the parts waiting to enter the stations.

Internal Successor

Opens the method which defines to which internal successor a part goes, meaning a part just entered the pivot, and this method decides to which station to turn.

Int. Successor Table

Opens the internal successor table.

External Successor

Opens the method which defines to which successor a part goes when it leaves the pivot.

Ext. Successor Table

Opens the external successor table.

The Lift Objects

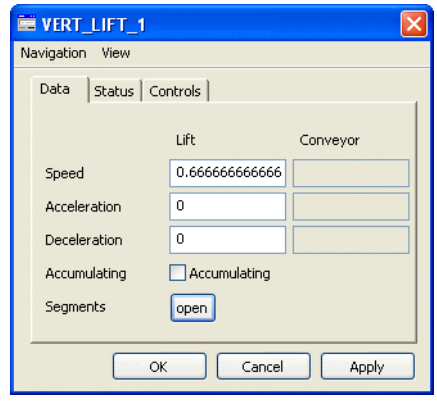
The lift has also two different sets of objects, one for floorconveyor and one for electrified monorail. A set of lift objects contains the control object, the lift track object and the stations. If a part wants to enter a station of a lift, the control object is called, which tells the lift object to move to the station. When the lift object arrives, the part can enter the station.

The Lift as a Part of a Floorconveyor

The stations are active objects, which means they transport the part in the lifter and out of the lifter.

The Control Object

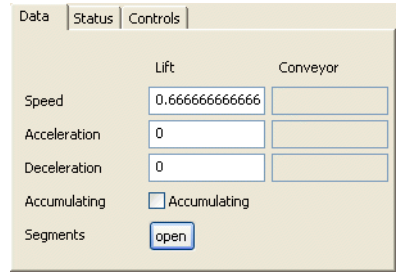
In the control object for the lift all information about the lift is given, like speed, status and the controls.



Picture 53:

Tab Data

On this tab all information concerning the lift is gathered.



Picture 54:The data tab of a conveyor lift control object

Lift Speed, Acceleration and Deceleration

The column contains the data for the lift behavior. Speed is the maximum speed, acceleration and deceleration define if the vertical motion is an accelerated movement or not. An acceleration value of zero means no acceleration.

Accumulating

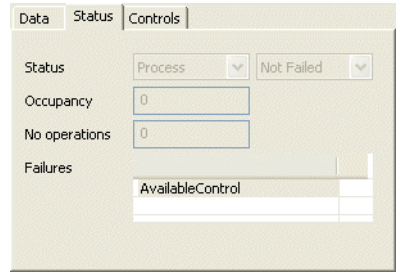
This value corresponds to the same values of the conveyor control object.

Segment Table

A table with all segment objects in it for random access.

Tab Status

On this tab all information concerning the state of the lift is gathered.



Picture 55:The status tab of a conveyor lift control object

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Number of Operations

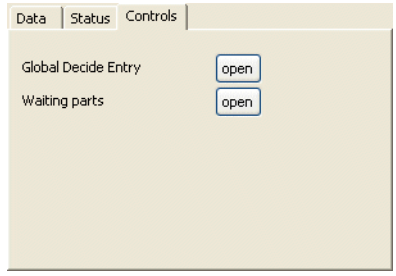
Number of parts that were on the segments.

Failures

A link to the failure generators of the object.

Tab Controls

The methods and tables concerning the material flow are available on this tab.



Picture 56:The controls of a conveyor lift control object

Global Decide Entry

This method decides which part enters the lift, when there is more than one part at the entrance stations.

Waiting Parts

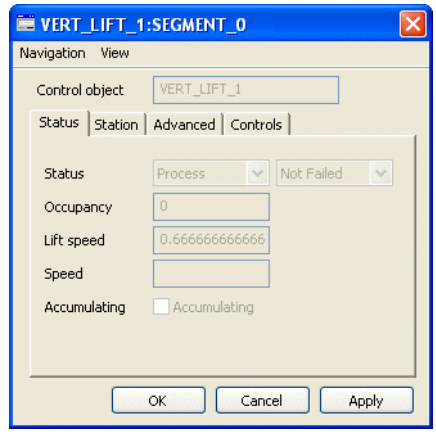
Opens the table with the parts waiting to enter the stations.

The Basic Segment Object

The basic segment is a normal Plant Simulation basic object of type track and has no additional dialog.

The Station

The station is a combination of the station segment and the station object itself which share the same dialog, similar the other stations.



Picture 57:A dialog of a conveyor lift station

Special Menus in the Conveyor Lift Station Segment

In addition to the normal menus this segment has the menu items Contentlist and Control object.

Menu Item Contentlist

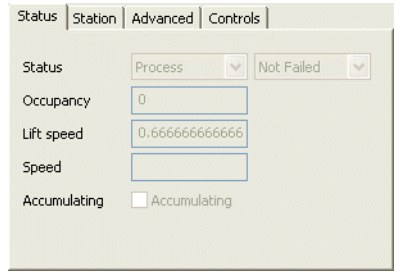
Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.



Picture 58:The status tab of a conveyor lift station

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Lift Speed

The actual speed of the lift object.

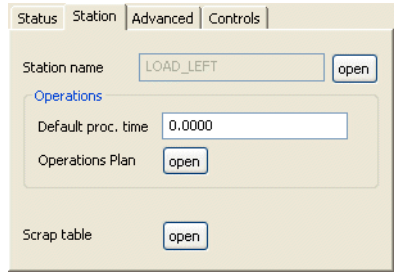
Accumulating

This value corresponds to the same values of the conveyor control object.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

Tab Station

On this tab all information about the station is assembled.



Picture 59:The station tab of a conveyor lift station

Station Name

The name of the station, as given in FactoryCAD.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

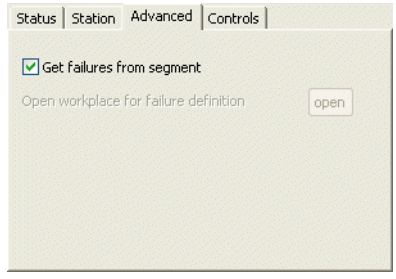
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 60:The advanced tab of a conveyor lift station

Get Failures from Segment

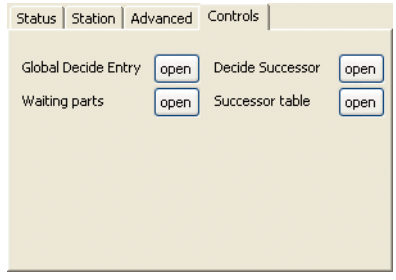
Normally, a station is failed when the object is failed. If you want to break this connection, then you have to uncheck this box.

Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

The methods and tables concerning the material flow are available on this tab.



Picture 61:The controls of a conveyor lift station

Global Decide Entry

This method decides which part enters the lift, when there is more than one part at the entrance stations.

Waiting Parts

Opens the table with the parts waiting to enter the stations.

Decide Successor

Opens the method which defines to which successor a part goes.

Successor Table

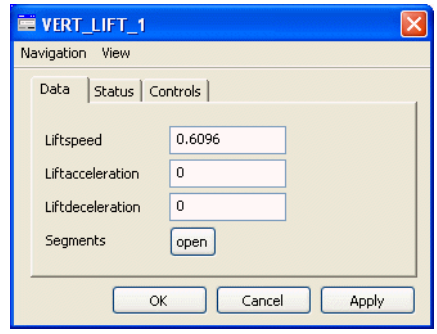
Opens a table with all the possible successor objects.

The Lift as a Part of a Electrified Monorail

The stations are just control objects, the part, normally a monorail transporter, drives on the lift by itself.

The Control Object

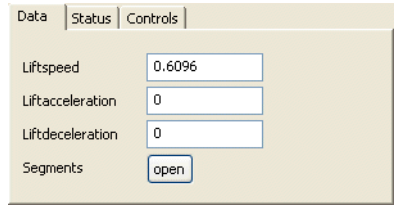
In the control object for the lift all information about the lift is given, like speed, status and the controls.



Picture 62:

Tab Data

On this tab all information concerning the lift is gathered.



Picture 63:The data tab of a electrified monorail lift control object

Lift Speed, Acceleration and Deceleration

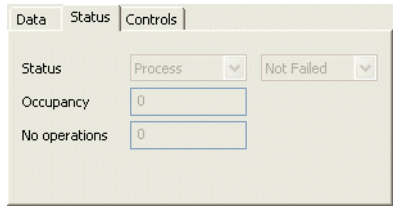
The column contains the data for the lift behavior. Speed is the maximum speed, acceleration and deceleration define if the vertical motion is a accelerated movement or not. An acceleration value of zero means no acceleration.

Segment Table

A table with all segment objects in it for random access.

Tab Status

On this tab all information concerning the state of the lift is gathered.



Picture 64:The status tab of a electrified monorail lift control object

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

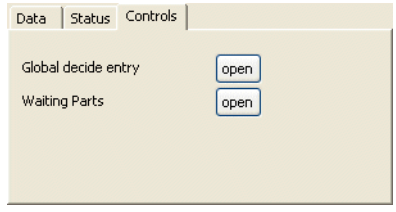
The field Occupancy shows how many parts are located on the segment.

Number of Operations

Number of parts that were on the segments.

Tab Controls

The methods and tables concerning the material flow are available on this tab.



Picture 65:The controls of a conveyor lift control object

Global Decide Entry

This method decides which part enters the lift, when there is more than one part at the entrance stations.

Waiting Parts

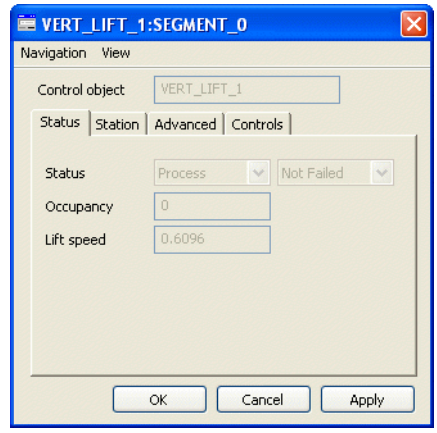
Opens the table with the parts waiting to enter the stations.

The Basic Segment Object

The basic segment is a normal Plant Simulation basic object of type track and has no additional dialog.

The Station

The station is a combination of the station segment and the station object itself which share the same dialog, similar the other stations.



Picture 66:A dialog of a electrified monorail lift station

Special Menus in the Station Dialog

In addition to the normal menus the station has the menu items Contentlist and Control object.

Menu Item Contentlist

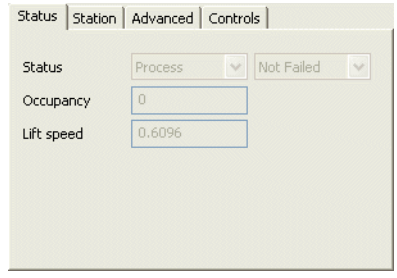
Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.



Picture 67:The status tab of a conveyor lift station

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

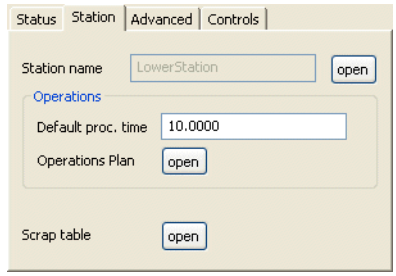
The field Occupancy shows how many parts are located on the segment.

Lift Speed

The actual speed of the lift object.

Tab Station

On this tab all information about the station is assembled.



Picture 68:The station tab of a electrified monorail lift station

Station Name

The name of the station, as given in FactoryCAD.

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

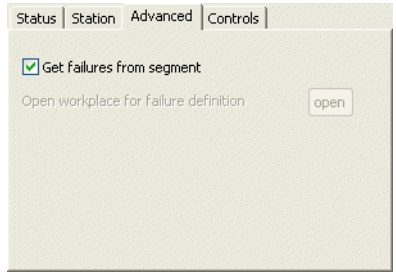
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 69:The advanced tab of a electrified monorail lift station

Get Failures from Segment

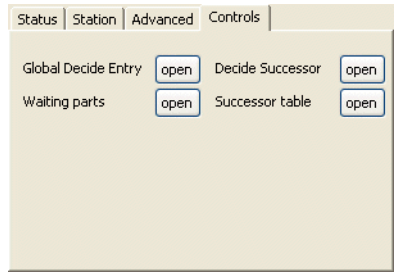
Normally, a station is failed when the object is failed. If you want to break this connection, then you have to uncheck this box.

Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

The methods and tables concerning the material flow are available on this tab.



Picture 70:The controls of a electrified monorail lift station

Global Decide Entry

This method decides which part enters the lift, when there is more than one part at the entrance stations.

Waiting Parts

Opens the table with the parts waiting to enter the stations.

Decide Successor

Opens the method which defines to which successor a part goes.

Successor Table

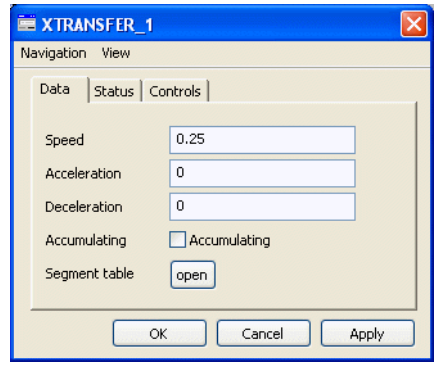
Opens a table with all the possible successor objects.

The Crosstransfer Objects

The crosstransfer is used to transport loads or carriers sideways without changing their alignment. It is usable only in the context of floorconveyors.

The Control Object

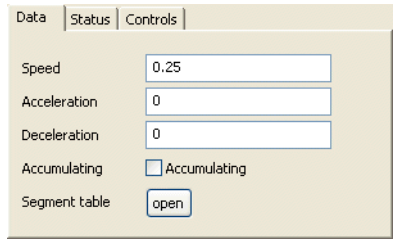
In the control object all relevant information about the behavior of the crosstransfer is presented.



Picture 71:The dialog of a crosstransfer control object

Tab Data

On this tab all information about the speed of the crosstransfer is given.



Picture 72:The data tab of the crosstransfer control object

Speed

The maximum speed of the cross segment.

Acceleration and Deceleration

Defines if acceleration and deceleration is used. A value of zero in acceleration means it is not used.

Accumulating

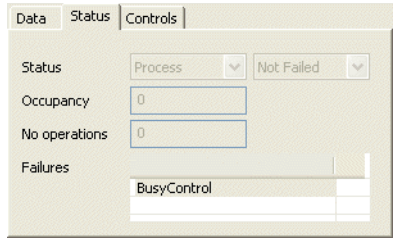
If the cross segment is rigid, uncheck the checkbox, otherwise the crosssegment accumulates packages on itself.

Segment Table

A table with all segment objects in it for random access.

Tab Status

On this tab the status of the crosstransfer is shown: Paused or Failed, occupancy and number of operations.



Picture 73:The status tab of a crosstransfer control object

Status

If the crosstransfer is available, paused or between shifts, and if it is failed or not.

Occupancy

How many parts are on the crosstransfer

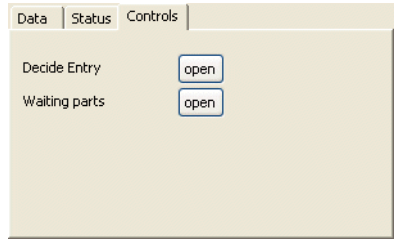
Number of Operations

Number of parts that were on the crosstransfer

Failures

A link to the failure generators of the crosstransfer.

Tab Controls



Picture 74:The controls tab of a crosstransfer control object

Decide Entry

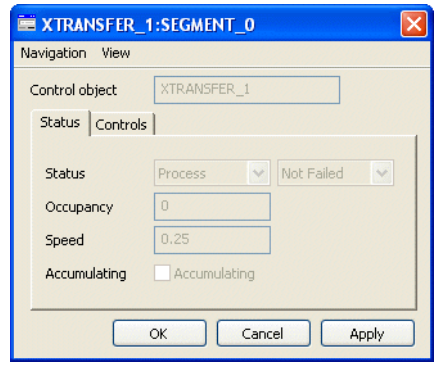
This button opens a method which decides which part will enter the crosstransfer.

Waiting Parts

Opens a table with all parts waiting to be transported with the crosstransfer.

The Basic Segment Object

This segment just transports the parts sideways.



Picture 75:The dialog of a basic segment of a crosstransfer

Special Menus in the Crosstransfer Basic Segment

In addition to the normal menus the basic segment has the menu items Contentlist and Control object.

Menu Item Contentlist

Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Speed

The actual speed of the conveyor.

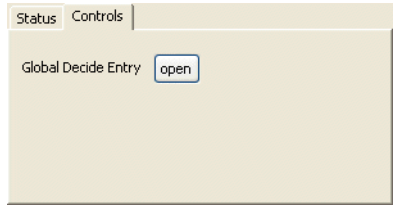
Accumulating

If the conveyor is rigid, the checkbox is unchecked, otherwise the conveyor accumulates packages on itself.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

Tab Controls

On the basic segment only the method for the global entry is linked.



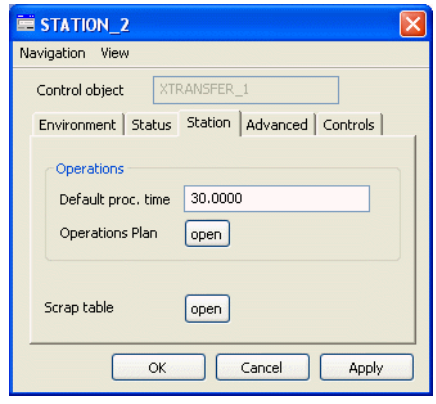
Picture 76:The status tab of a pivot segment

Global Decide Entry

This method decides which part enters the crosstransfer, when there is more than one part waiting for transport.

The Station

The crosstransfer station is unique because it consists of four objects: the station object, the station segment, cross segment left and cross segment right. A part enters the station on a cross segment, is then moved to the station segment and then transported via the basic segments of the crosstransfer. A part leaves with entering the station segment, is moved to the corresponding cross segment and from there it moves on to the next object (it does not matter if the next object is a conveyor or any other object).



Picture 77:The dialog of a crosstransfer station

Special Menus in the Crosstransfer Station Segment

In addition to the normal menus this segment has the menu items Contentlist and Control object.

Menu Item Contentlist

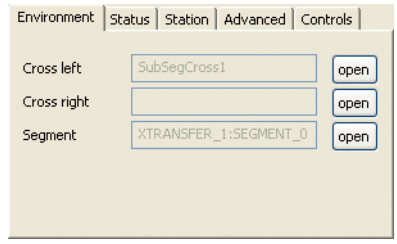
Opens a table which shows the content of the segment.

Menu Item Control Object

Opens the dialog of the control object of the segment.

Tab Environment

This tab allows access to all segments of the station.



Picture 78:The environment tab of the crosstransfer station

Cross Left

This segment is used when a parts enters from the left or leaves to the right.

Cross Right

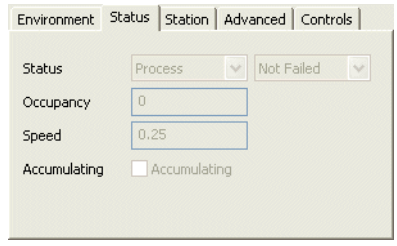
This segment is used when a parts enters from the right or leaves to the left.

Segment

This is the station segment, which transports the parts sideways.

Tab Status

This tab shows the status of the segment: Paused or Failed, occupancy and speed.



Picture 79:The status tab of the crosstransfer station

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

The field Occupancy shows how many parts are located on the segment.

Speed

The actual speed of the conveyor.

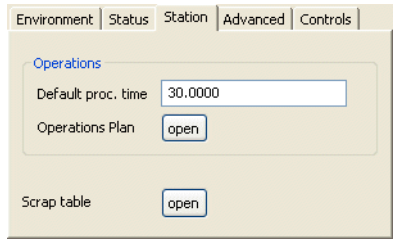
Accumulating

If the conveyor is rigid, the checkbox is unchecked, otherwise the conveyor accumulates packages on itself.

Note: You cannot change this data here, it is only possible to change data concerning speed in the control object!

Tab Station

This tab has all the information about the behavior of the station.



Picture 80:The tab station of the crosstransfer station

Default Proc Time

The default processing time, is used if the process time is not given in the operation plan.

Operations Plan

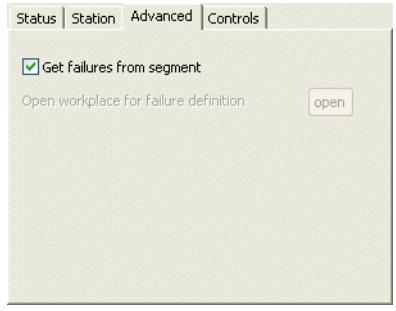
Opens the operation plan table which can contain a part specific procedure time, set-up time, required parts for assembly operations etc.

Scrap Table

This table contains all parts which became scrap at this station.

Tab Advanced

This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 81:The advanced tab of a crosstransfer station

Get Failures from Segment

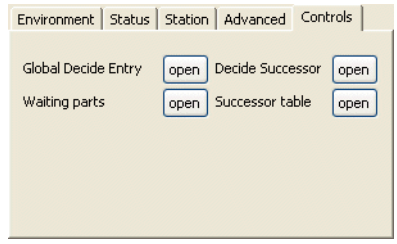
Normally, a station is failed when the transport device is failed. If you want to break this connection, then you have to uncheck this box.

Open Workplace for Failure Definition

This dialog element is only available if the checkbox is unchecked. The button opens the Plant Simulation objects which represents the core station. In this Plant Simulation object you can define the failure times for the station. (Look at the eM-Plant documentation for more details how to set failure times.)

Tab Controls

Contains the methods and tables for controlling the entry and exit of parts on the crosstransfer.



Picture 82:The tab controls

Global Decide Entry

This method decides which part enters the crosstransfer, when there is more than one part waiting for transport.

Waiting Parts

The button opens a table which contains all parts waiting for transport on the crosstransfer.

Decide Successor

Opens the method which defines to which successor a part goes.

Successor Table

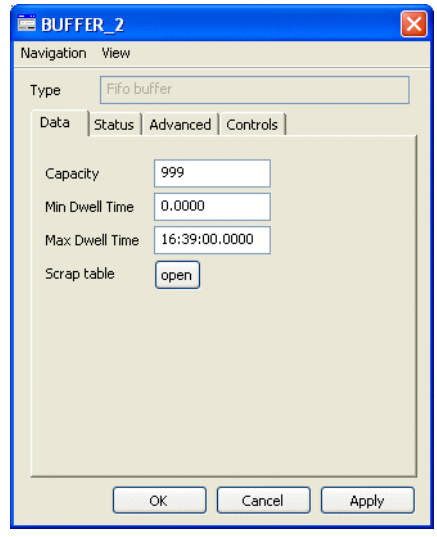
Opens a table with all the possible successor objects of this station.

The Resource Objects

The resource objects are place-oriented objects, meaning that they have a capacity independent of the size of the moveable units.

The Buffer Objects

All buffer objects share the same properties, they differ only in the sequence they move their content.



Picture 83:Dialog of a standard buffer

Special Menus in the Buffer Dialog

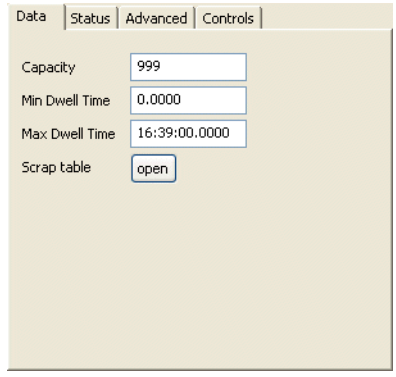
In addition to the normal menus the buffer dialog has the menu item Content.

Menu Item Content

Opens a table which shows the content of the buffer.

Tab Data

The tab contains data like capacity and the dwell times.



Picture 84:The tab Data

Capacity

Defines how many parts can be stored in the buffer.

Min Dwell Time

When a part enters the buffer, it can only leave it after the minimal dwell time.

Max Dwell Time

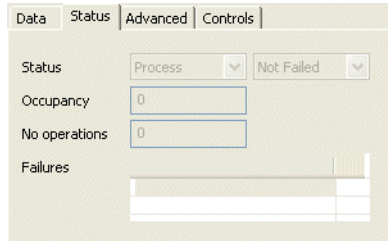
If a part is still on the buffer after the maximal dwell time it is deleted. A maximal dwell time of 0 means it can dwell on the buffer forever.

Scrap Table

If a part is scrapped, it is entered here.

Tab Status

Contains the data all resource objects share: status of the object, and links to the failure controls.



Picture 85:The tab Status of a buffer object

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

How many parts are on the buffer.

Number of Operations

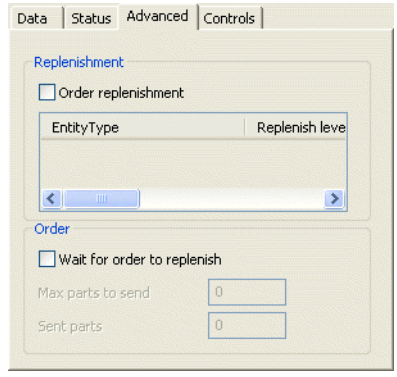
Number of parts that were on the buffer.

Failures

A link to the failure generators of the object.

Tab Advanced

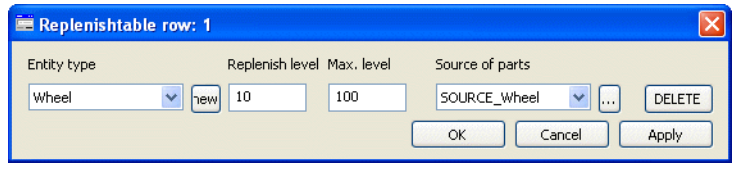
This tab contains advanced features which are not part of the standard dialogs in FactoryCAD.



Picture 86:The tab Advanced of a buffer

Replenishment

If you activate replenishment, the buffer will order the supplies you have entered in the table. Double clicking a line on the table opens the Edit dialog.



Picture 87:The Edit dialog of the replenish table

Entity Type

In the drop down menu the program fills automatically all entity types which are defined in the successor table. With the button new you can define new entity types.

Replenish Level

If the stock of the defined entity type is lower than the replenish level, a replenish order will be sent to the source.

Max. Level

Up to this amount a replenish order will be sent

Note: So the ordered amount will be the maximum stock minus the actual stock.

Source of Parts

The system will try to fill the drop down list with sources for the entity type, if this won´t work you have to look for the source by hand. It is also possible to have another buffer as a source of parts, this way whole sequences of buffers can be modelled.

Delete

With this button you can delete a line in the replenish table.

Order

If you check this, the buffer won´t move its content, it will wait for an order from a successor station (e.g. another buffer with Order Replenishment)

Max Parts to Send

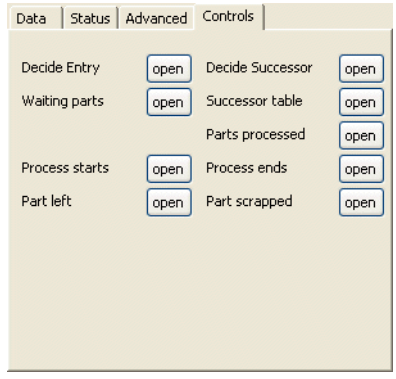
This is the threshold how many parts have to be sent to the successors.

Sent Parts

This is the number of parts already sent to the successors.

Tab Controls

On this tab the two user control methods for entrance and exit are available.



Picture 88:The tab Controls of a buffer

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance.

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the object.

Successor Table

Opens a table with all the possible successor objects.

Parts Processed

Opens the table which contains all parts waiting for transport and their designated target.

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

This method is called when a part is scrapped on the station.

The FIFO Buffer

This type of buffer is the one of the two standard buffer. The first to go in is the first to go out.

The LIFO Buffer

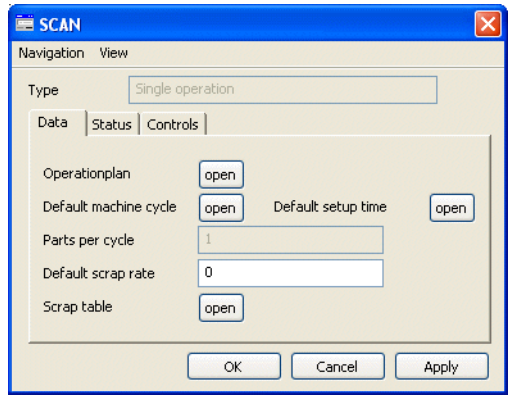
The second of the two standard buffer types. The last part to go in is the first part to go out.

The RestoreSequence Buffer

A third useful type of buffer used to restore the sequence of parts. Every time a part enters the buffer, the parts on it are sorted and the oldest part goes out first.

The Machine Objects

There are different types of machines, but all share a similar dialog.

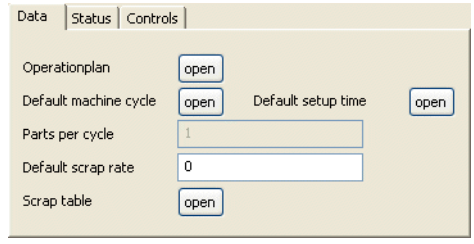


Picture 89:A single operation machine

A machine has no additional menu item.

Tab Data

All information concerning the processing of a part on a machine is on the tab Data of the machine dialog.



Picture 90:The data tab of a machine

Operationplan

Opens a table which contains the operation plan.

Default Machine Cycle

In this variable the default machine cycle is stored. This time is used if nothing is given in the operation plan.

Default Setup Time

Another variable contains the default setup time. This time is used if nothing is given in the operation plan.

Parts per Cycle

How many parts are processed in one machine cycle. (Only used in BatchOperation and Assembly machines)

Default Scrap Rate

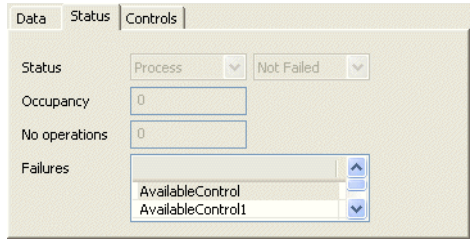
Defines how many parts are scrapped if nothing is given in the successor tables.

Scrap Table

When a part is scrapped, it is noted down here.

Tab Status

As all other objects, the status of the machine. If it is paused, failed, or between shifts.



Picture 91:The tab Status of a machine with at least two failure generators

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

How many parts are on the buffer.

Number of Operations

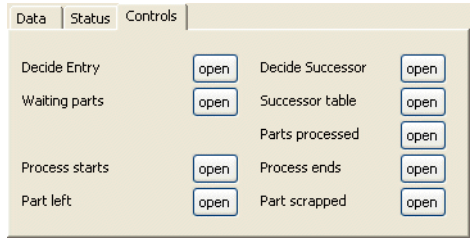
Number of parts that were on the buffer.

Failures

A link to the failure generators of the object.

Tab Controls

On this tab the two user control methods for entrance and exit are available.



Picture 92:The tab Controls of a machine

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance.

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the object.

Successor Table

Opens a table with all the possible successor objects.

Parts Processed

Opens the table which contains all parts waiting for transport and their designated target.

Process Starts

This method is called when the station begins to process a part.

Process Ends

This method is called when the station finished processing a part.

Part Left

This method is called when a part left the station.

Part Scrapped

This method is called when a part is scrapped on the station.

The Single Operation Machine

The single operation machine has a capacity of one, processes one part per cycle and the part must be moved before the next part can be processed.

The Batch Operation Machine

The batch operation machine processes a batch, that the capacity is equal to the parts per cycle setting. The machine needs a full batch to begin processing, and each full batch must be moved before the next can be processed.

The Multistation Machine

A multistation machine consists of several separated stations. Each station has a capacity of one, and acts independently from the others.

The Production Machine

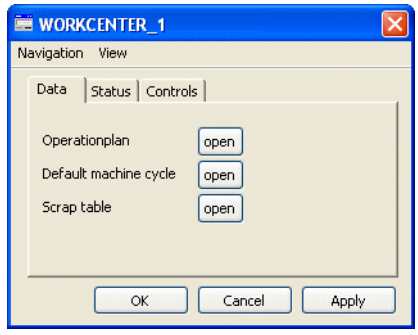
The production machine produces parts, one comes in, several go out. Only one part per cycle goes in, the next part can only enter when all parts of the last production cycle are moved.

The Assembly Machine

The assembly machine combines parts to a new part, it can also process several parts per cycle.

The Workcenter Object

The workcenters are simple workplaces, which cannot be failed and lack some parameters of the machines.

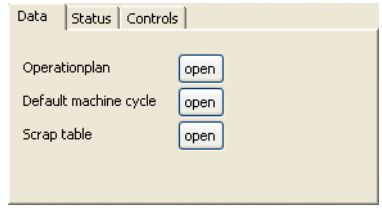


Picture 93:The dialog of a workcenter

A workcenter has no additional menu items.

Tab Data

All information concerning the processing of a part on a workplace is on the tab Data.



Picture 94:The data tab of a machine

Operationplan

Opens a table which contains the operation plan.

Default Machine Cycle

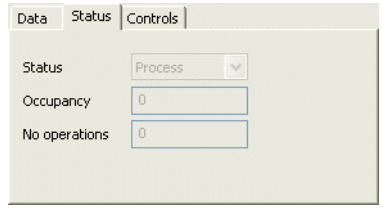
In the this variable the default machine cycle is stored. This time is used if nothing is given in the operation plan.

Scrap Table

When a part is scrapped, it is noted down here.

Tab Status

As all other objects, the status of the machine. If it is paused, failed, or between shifts.



Picture 95:The tab Status of a workplace

Status

If the object is available, paused or between shifts.

Occupancy

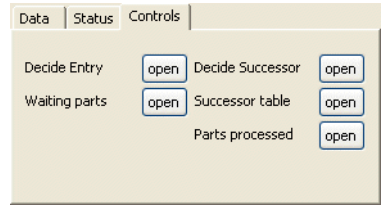
The field Occupancy shows how many parts are located on the segment.

Number of Operations

Number of parts that were on the segments.

Tab Controls

On this tab the two user control methods for entrance and exit are available.



Picture 96:The tab Controls of a workplace

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance.

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the object.

Successor Table

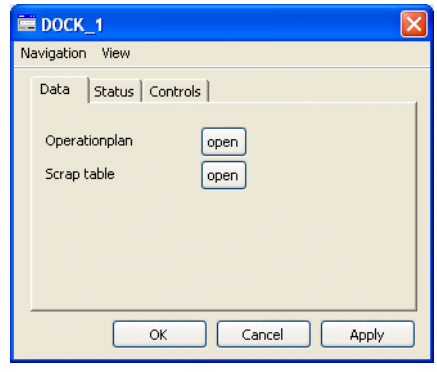
Opens a table with all the possible successor objects.

Parts Processed

Opens the table which contains all parts waiting for transport and their designated target.

The Dock Objects

The dock objects are the entrances and exits of the system. The docks are different in real structure, but are identical in behavior concerning simulation.

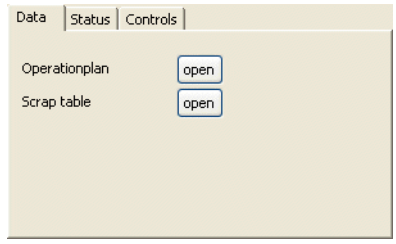


Picture 97:The dialog of a Dock object

A dock has no additional menu item.

Tab Data

All information concerning the processing of a part on a workplace is on the tab Data.



Picture 98:The data tab of a dock

Operationplan

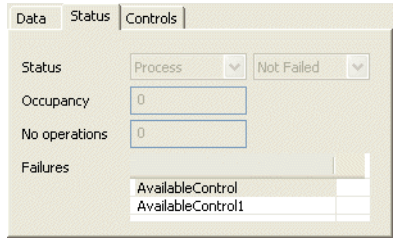
Opens a table which contains the operation plan.

Scrap Table

When a part is scrapped, it is noted down here.

Tab Status

As all other objects, the status of the machine. If it is paused, failed, or between shifts.



Picture 99:The tab Status of a dock with two failure generators

Status

The status field shows if the object is waiting, working, paused, unplanned (meaning it's not shift time), failed or blocked.

Occupancy

How many parts are on the dock.

Number of Operations

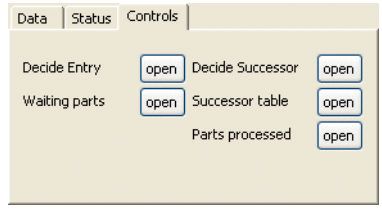
Number of parts that were on the dock.

Failures

A link to the failure generators of the object.

Tab Controls

On this tab the two user control methods for entrance and exit are available.



Picture 100:The tab Controls of a dock

Decide Entry

This method decides which part enters the segment, when there is more than one part at the entrance.

Decide Successor

Opens the method which defines to which successor a part goes.

Waiting Parts

Opens the table with the parts waiting to enter the object.

Successor Table

Opens a table with all the possible successor objects.

Parts Processed

Opens the table which contains all parts waiting for transport and their designated target.

The Types of Docks

There are three different types of docks: EndUnload dock, SideUnload dock and Internal dock. They are different in geometrical sense, how the external transporter (van or truck) connects to the dock.

Concerning the simulation, they are identical.

The Robot Objects

The robot objects share the simulation behavior with the machines, only the graphical display is different.

Note: For description of the dialog, look on the description for machine objects

The Standard Robot

The standard robot is based on the behavior of the single operation machine and shares all behavior of that object.

The Material Handling Robot

The material handling robot is also based on the behavior of the single operation machine and shares all behavior of that object.

The Assembly Robot

The material handling robot is also based on the behavior of the assembly machine and shares all behavior of that object.

The Personal Objects

The personal objects are the basic objects of Plant Simulation, for further details consult the Plant Simulation reference manual.

The Operator

The operator is simulated with the exporter basic object of Plant Simulation. For further details consult the Plant Simulation reference manual.

The Broker

The broker is used to distribute the operators to the stations which request services for processing or setup. For further details consult the Plant Simulation reference manual.

The Load Object

The load object represents all parts which are processed in the system. For the load object we use the Plant Simulation basic object entity with some additional attributes.

Additional Attributes

Some custom attributes are used for the system, but can be used additionally for customizing a simulation model.

EntityType

In this attribute of type string the system writes the name of the load object. Use it to separate different types of loads.

ObjHeight, ObjLength, ObjWidth

In this attribute of type real the system stores the height, width and length of the load.

ProcessedBy

In this attribute of type object (called object link) the system writes the station or object which processes that load at the moment. If it is empty (void) then the load is on transport.

Target

Target is an attribute of type object and can be used to override the normal percentage based exit behavior if the correct tables are filled.

OrderNo

This attribute of type string is not used by the system, it is free for you to customize the exit behavior of the system.

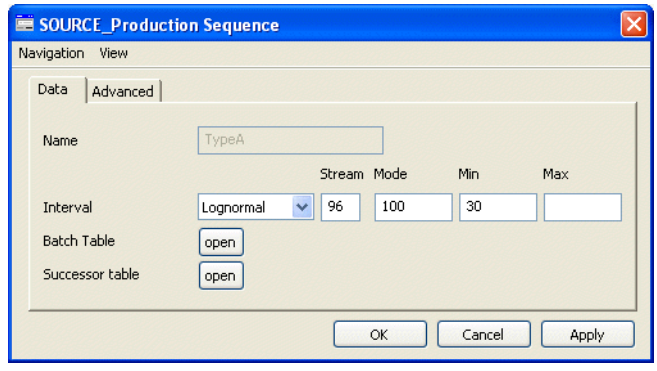
WaitingStations, PushStation

The variable WaitingStations is a table-attribute, the variable pushStation is a method-attribute, both are used by the system to synchronize the processing if a load wants to be processed by more than one station.

Example:
```
Two ProcessRun stations follow each other, and the transport distance between them
is not big enough for the processing time of the predecessor station, which means
that by the time it reaches the second station, it is still processed by the first
station.
```

The Source of Parts

For the creation of the loads a special object is needed. This object contains all data concerning the load and creates them in specific intervals. It also has some special abilities concerning the replenishment of buffers.

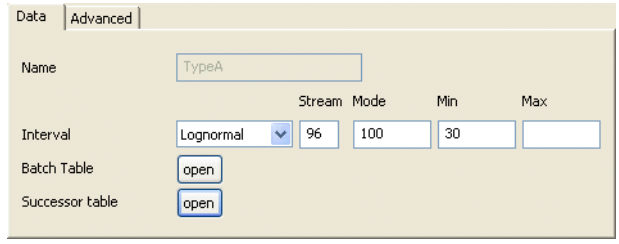


Picture 101:Dialog of a source for parts with a distribution as interval

A source object has no additional menus.

Tab Data

Contains all information about the parts and the link to the next object.



Picture 102:The data tab of the source dialog

Name

The name of the source, normally the name of the kind of load it will create.

Interval

The time between two creation processes. It can be a constant time or a distribution. For more information please look it up in the Plant Simulation user manual.

Batch Table

In the batch table it is possible to define the types of loads a source can create and the sequence in which it will create them.

	object 1	integer 2	string 3	table 4
string	MU	Number	Name	Attribute
1	.Application	20	TypeA	table41
2	.Application	10	TypeB	table42
3	.Application	30	TypeC	table43
4				

Picture 103:An example for a batch table where three different types of loads will be produced

MU

A link to the Plant Simulation basic moveable unit ("MU") which will be used for the different types of load objects.

Number

How many parts will be produced of one type of load before the source switches to the next type.

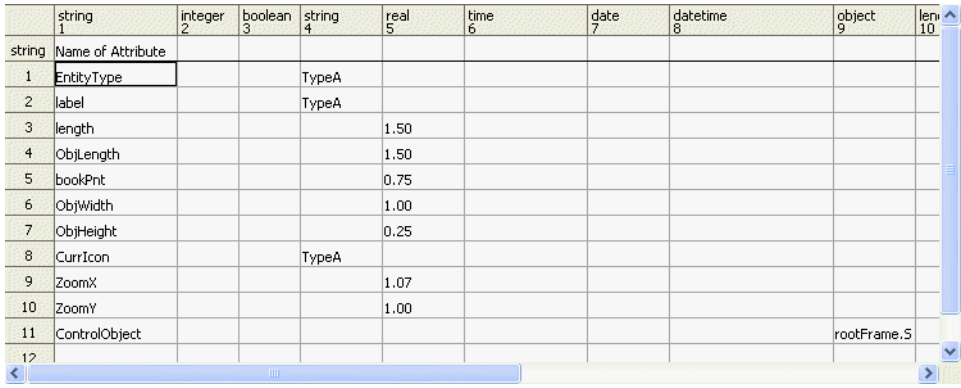
Example: `In the picture above, 20 loads of type TypeA, 10 loads of type TypeB and then 20 loads of type TypeC will be produced, then the sequence begins anew.`

Name

The name of the load to be produced

Attribute

In this subtable the attributes concerning the type of load are stored.



	string 1	integer 2	boolean 3	string 4	real 5	time 6	date 7	datetime 8	object 9	len 10
string	Name of Attribute									
1	EntityType			TypeA						
2	label			TypeA						
3	length				1.50					
4	ObjLength				1.50					
5	bookPnt				0.75					
6	ObjWidth				1.00					
7	ObjHeight				0.25					
8	CurrIcon			TypeA						
9	ZoomX				1.07					
10	ZoomY				1.00					
11	ControlObject								rootFrame.S	
12										

Picture 104:The attribute table of the load type TypeA

All attribute names are written in the first column, in the following columns the value of the attribute is given, each value in the column which has its value type.

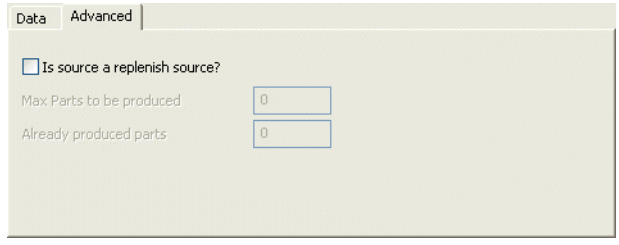
Example: `The attribute "EntityType" is of type "string", so the value of the attribute is in column number 4.`

Successor Table

Similar to the other objects, the source has also a successor table. Normally it should contain only one successor, but it can be changed.

Tab Advanced

On this tab it is possible to activate the advanced behavior of the source.



Picture 105:The tab Advanced of a source

If you activate this behavior, the source will only produce parts which are ordered by buffers or via information flow. These previously ordered parts will then be produced and transported by normal means.

Max Parts to Be Produced

This is the threshold how many parts have to be produced.

Already Produced Parts

This is the number of parts already produced.

The Carrier Objects

The carrier objects are represented by the Plant Simulation basic object container combined with some additional attributes and methods.

The Power and Free and Regular Monorail Carrier

In addition to the same attributes which are included in the loads all carriers have following attributes

EntityType

The load has also the attribute EntityType, but in case of carriers this attribute has a special meaning. Each load station sets this attribute to the same value as the load. This way the processing station will use the carrier the same way as a load entity of the same entity type.

ContainerType

The system writes the type of the carrier in this string-attribute. It can be used in user defined methods.

Path

In this attribute of type string the system writes the path of the carrier, usually the monorail system.

The Floorconveyor Container

There are two possibilities for floorconveyor containers:

1. A closed loop of conveyors. In this case the containers are created once and then just continue to move on the conveyors.
2. An open system of conveyors (where containers get back to the beginning of the conveyors some other way, e.g. by fork trucks). In this case the containers are created on the load stations and destroyed on the unload stations. So if you did not model the way back for the containers, they will just "fly" back to the load stations. Anyway, the total quantity of containers will not exceed the specified quantity.

The floorconveyor container has the same attributes as the other carriers and the following additional attributes:

Exitfreecounter, MovePart and GotPart

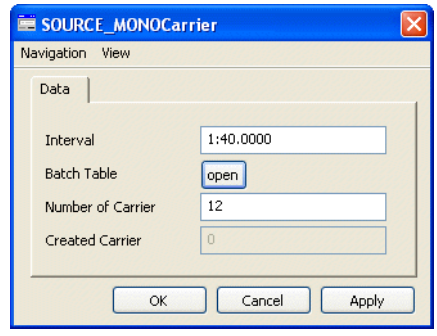
This attributes are used by the system to unload the parts from the container

LoadTimeContainer and UnloadTimeContainer

If there is no closed loop and the container are created and destroyed, this times are used to delay the creation or destruction of the containers. If a container is available, it will be created after the LoadTimeContainer, and if a container is emptied on a unload station it will wait for the UnloadTimeContainer on the station and will be destroyed afterwards.

The Source of Carriers

This object creates a predefined number of carriers with a predefined set of attributes similar to the source of parts.



Picture 106:Dialog of a source for carriers

A carrier source object has no additional menus.

Tab Data

Contains all information about the creation of carriers like interval, number of carriers and a button to open the attributes table.

Interval

The time between two creation processes. For more information please look it up in the Plant Simulation user manual.

Batch Table

In the batch table it is possible to define the types of carriers the source can create. In addition to the columns in the batch table for parts, a 5th column defining the shift calendar to which the carriers are assigned is added.

Number of Carrier

Contains the number of carriers you want to create.

Created Carrier

The number of carriers which are already created and in the system.

The Transporter Objects

The FactoryCAD transporter objects are represented by the Plant Simulation basic object transporter with some additional attributes.

The EMS Transporter

The electrified monorail system is a version of the regular monorail system where all transporters are independent, meaning that all transporter can drive independently from each other, have different failures and speed.

The Standard Transporter

The transporter can be an AGVS, a fork lift etc. It has a defined path, a speed, failure controls and a shift.

Attributes of all Transporters

Each transporter has a set of custom attributes, which define their behavior.

ControlObject

The attribute ControlObject is a link to the controlling object of the transporter. A detailed description of the ability of the control object follows.

EntityType

The load has also the attribute entity type, but in case of transporters this attribute has a special meaning. Each load station sets this attribute to the same value as the load. This way the processing station will use the transporter the same way as a load entity of the same entity type.

LoadTime, UnloadTime

This times are additionally to the times defined in the appropriate stations.

ObjHeight, ObjLength, ObjWidth

Defines the height, length and width of the transporter objects. Plant Simulation uses only the length of the transporters.

Path

In this string variable the systems stores which path the transporter is allowed to take. A track can belong to different paths, but each transporter belongs only to one path.

ProcessedBy

In this object link-attribute the system writes the station or object which processes that load at the moment. If it is empty (void) then the load is on transport.

WaitingStations

The variable WaitingStations is a table-attribute and is used by the system to synchronize the processing if a transporter wants to be processed by more than one station.

Target

In this variable the systems stores the next target of the transporter. Only this station can process the transporter and give him a new target.

Additional Attribute of the Standard Transporter

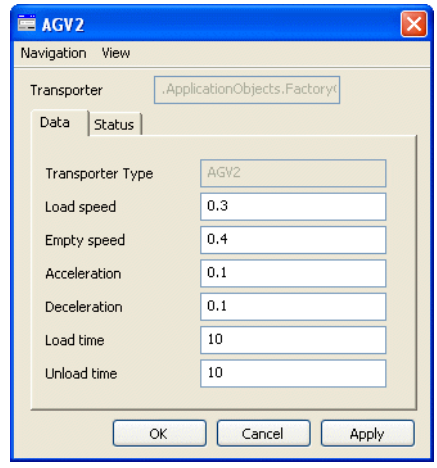
The standard transporter has an additional attribute which provides an additional ability of the standard transporter: it can drive to a "home" station if there is no order waiting for it.

WaitforOrder

This object link stores the waiting station for the transporter. Each standard transporter asks for an order when it is emptied, when there is no order, it will drive to the waiting station, if there is no station given in the variable, it will drive aimlessly around.

The Transporter Control Object

The control object is responsible for creating failures, managing speed and acceleration and the state of the transporter.

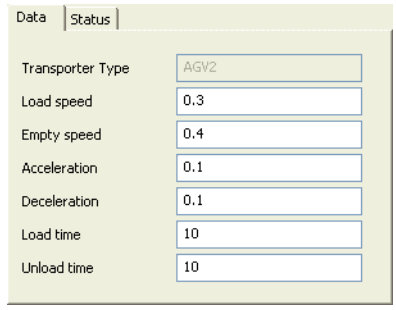


Picture 107:Dialog of an AGV

The transporter control object has one additional menu: Transporter. It opens the corresponding transporter object.

Tab Data

All information concerning the behavior of the transporter except the failure definition is on this tab.



Picture 108:The data tab of the dialog

Transporter Type

Defines the transporter type of the object. This can´t be changed during runtime.

Load Speed

The speed of the transporter if it is not empty.

Empty Speed

Each Unload station checks if the transporter is empty and sets the speed to empty speed when it is done.

Acceleration

Defines an acceleration value of the transporter. A value of zero means there is no acceleration, the speed changes immediately.

Deceleration

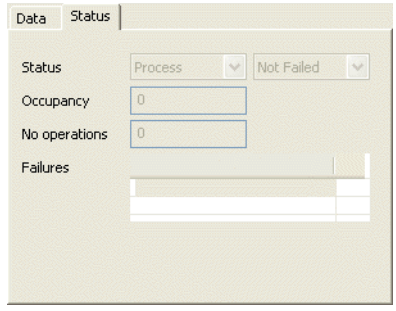
Defines the deceleration value of the transporter.

Load Time, Unload Time

These times are in addition (!) to the load/unload times defined in the appropriate stations.

Tab Status

On this tab is all information about the status of the transporter, if it is failed or paused, occupied or empty.



Picture 109:The status tab of the dialog

Status

If the transporter is available, paused or between shifts.

Occupancy

How many parts are on the transporter.

Number of Operations

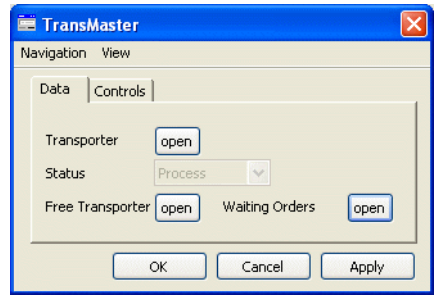
Number of parts that ever were on the transporter.

Failures

A link to the failure generators of the transporter.

The Transporter Master Control

In this control uses all transporter controls. It also contains the control strategies for allocating transporters to transport orders.

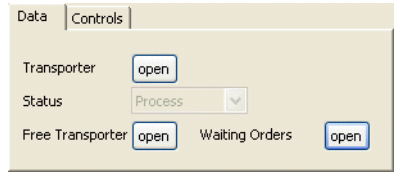


Picture 110:The dialog of the Transporter Master Control

A Transporter Master Control has no additional menus.

Tab Data

This tab offers links to different tables and shows the status of all transporters.



Picture 111:The data tab of the Transporter Master Control dialog

Transporters

The button opens a table which contains all transporter control objects and transporters.

Status

If the transporters are available, paused or between shifts.

Free Transporters

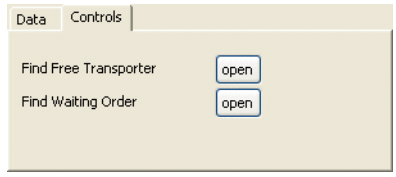
Opens a table containing all transporter which have no load on it. You can also see if a transporter is reserved for a specific station.

Waiting Orders

In this table you can see if a station is waiting for a transporter and could not reserve one etc.

Tab Controls

On this tab you can find links to two methods which control the transporter system.



Picture 112:The controls tab of the Transporter Master Control dialog

Find Free Transporter

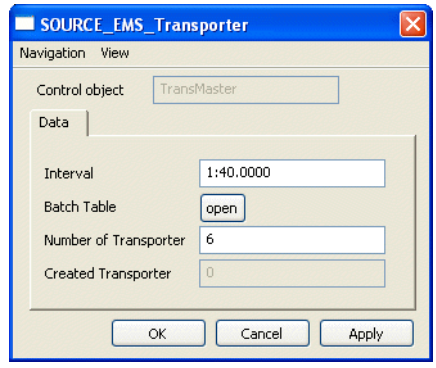
The button opens a method which gets a list of transporters which can fulfill the transport order.

Find Waiting Order

The method is called whenever a transporter is unloaded. It is used to give the transporter a new transport order if a station is waiting for transport.

The Source of Transporters

This object creates a predefined number of transporters with a predefined set of attributes and links them to their control object.



Picture 113:Dialog of a source for transporters

A transporter source object has a new menu under View: Control Object. It opens the Transporter Master Control object of the model.

Tab Data

Contains all information about the creation of transporters like interval, number of transporters and a button to open the attributes table.

Interval

The time between two creation processes. For more information consult the Plant Simulation user manual.

Batch Table

In the batch table it is possible to define the types of carriers the source can create. In addition to the columns in the batch table for parts, a 5th column defining the shift calendar to which the carriers are assigned is added.

	object 1	integer 2	string 3	table 4	object 5	object 6	object 7
string	MU	Number	Name	Attribute	Control	Home Station	Shift calendar
1	.Application	1	AGV2	table41	TransporterCtrl	~.load	~.ShiftCalendar1
2	.Application	1	AGV2	table41	TransporterCtrl	~.Load1	~.ShiftCalendar1
3	.Application	1	AGV2	table41	TransporterCtrl	~.load	~.ShiftCalendar1
4	.Application	1	AGV2	table41	TransporterCtrl	~.Load1	~.ShiftCalendar1
5							
6							

Picture 114:Batch table of a AGV transporter source

Control

In this column there must be a control object for each transporter. This control object will be copied to the Transporter Master Control object and used during the simulation.

HomeStation

Use this field to enter a home station. If a transporter has nothing to do (no waiting transport) it will drive to the home-station, usually a load station. This object must be a station which is reachable for the transporter (meaning it's located on a path where the transporter can drive and there's a way to the station the transporter can use).

Shift Calendar

In the last column it is possible to enter a shift calendar for each transporter. In the picture above it is the same for each transporter.

Number of Transporters

Contains the number of transporters you want to create.

Created Carrier

The number of transporters which are already created and in the system.

Administrative Objects

These are objects for administration and control, which are not included in the material flow.

Failure Definition

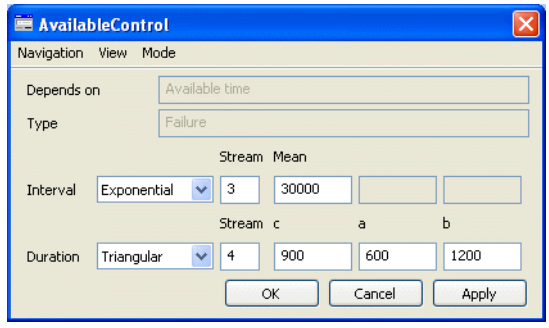
There are three different kinds of failures: Available time dependant, busy time dependant and number of operations dependant. It is possible for one object to have more than one failure definition attached, e.g. a driller bit that breaks every 1000th operation combined with a failure due electrical breakdown of the machine every two weeks.

Available Time Dependant

A failure can occur if the object is available, meaning not in pause or between shifts.

Definition of the Times

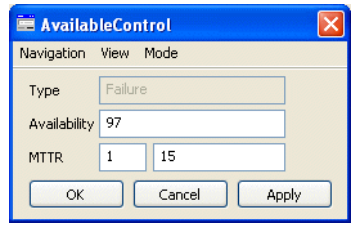
The `interval` is the time *Mean Time Between Failure* and `duration` is the time *Mean Time To Repair*.



Picture 115:A dialog of a available time dependant failure definition

It is possible to define distributions for the time values: `Constant`, `Exponential`, `Triangular`, `Lognormal`, `Weibull` and `Uniform`.

For more information about distribution and times consult the Plant Simulation manual.

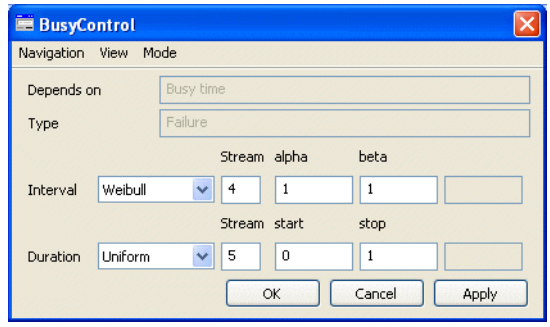


Picture 116:Alternative dialog of a available time dependant failure definition

It is also possible to define the combination of `Availability` and `MTTR`. To change between the two modes, please go to the mode-menu and change to the other mode.

Busy Time Dependant

A failure can occur if the object is working, meaning not in pause or between shifts. A failure cannot occur if the object is failed by another failure definition.

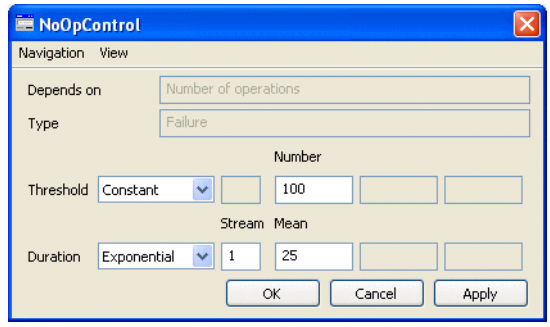


Picture 117:A dialog of a busy time dependant failure definition

The busy time dependant failure definition has the same possibilities as the available time dependant failure definition.

Number of Operations Dependant

This failure definition is used to simulate failures that occur after a certain number of operations, e.g. the breaking of a driller bit.



Picture 118:A dialog of a number of operations dependant failure definition

The `threshold` defines the number of operations after that this failure occurs, the `duration` defines how long this failure occurs. It is possible for both values to be distributions.

Shift Definition

The shift calendar object includes the definition of pauses, when shifts end and start. It utilizes the Plant Simulation basic object ShiftCalendar. Consult the Plant Simulation manual for further information.

Working with the Model

In this chapter we describe how to change the behavior of the model after creation. There are user-definable methods and tables in all objects which change the material flow. The most important are:

- SuccessorTable
- DecideSuccessor
- DecidePart
- OperationPlan

Successor Table

In this table the system defines which other objects are successors of the object where the part is located at the moment. The first level separates the different part types, so that each part type (attribute EntityType) has a different route.

	string 0	table 1
string	Part	Successordata
1	TypeB	table11
2	TypeC	table12
3		
4		

Picture 119:A successor table with two parts

The next level is the successor level, there can be different successors given for each part.

	object 0	integer 1	string 2	integer 3	real 4
string	Successor	Probability	Kind of Movement	Movement quantity	Scrap Rate
1	.SDXModels.Lift_Part2.	50	Smooth	1	0.00
2	.SDXModels.Lift_Part2.	50	Smooth	1	0.00
3					

Picture 120:The successordata subtable of one part

Each possible successor is defined in one line of this subtable.

Successor

In the first (the row index) column the location of the successor is given. This column is of the type “object” and is a direct link to the successor object.

Probability

The probability values of the column must add up to 100%. If it adds up to more then 100%, the system only works with the rows that add up to 100% and lower.

Kind of Movement

There are two possibilities: “Smooth” and “Place”. Smooth means that a part is moved on the same pathway, e.g it stays on the floorconveyor system. Place means that it leaves the system, e.g. from a floorconveyor on a machine.

Movement Quantity

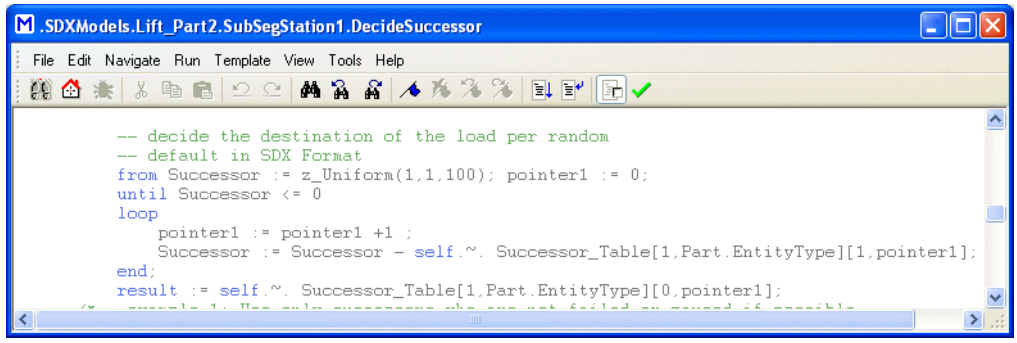
How many parts are moved at the same time. Normally one, but some exceptions exist.

Scrap Rate

On each transport operation there is a probability for failure, e.g. a part fall off a container. With this value the probability of such a failure is given.

DecideSuccessor

This method is called when a part tries to leave an object. It gets the part as argument and returns the successor object. This method is user-defined, which means you can change it to suit your need. The default setting is that it looks into the successor table and distributes the parts according to the list.



Picture 121:A sample DecideSuccessor method

As long as the return-value is a regular successor of the object the method is valid, you do not have to use a distribution for it, e.g. test for the status of the successor or the occupancy of object further down the line.

DecidePart

This method is called when a part want to enter a object and the object is able to get a part. The method gets the WaitingParts list as parameter, and the return value is the part chosen for entrance or void if no part should enter the object.

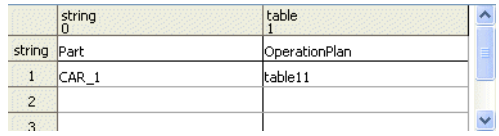
```
(WaitingParts : table):object
is
    found    : boolean;
    i        : integer;
do
    i := 1; found := false;
    while not found and i <= WaitingParts.ydim loop
        if WaitingParts[1,i].ExitFreeCounter = 0 then
            result := WaitingParts[0,i];
            found := true;
        else
            i := i+1;
        end;
    end;

end;
```

Picture 122:A sample method text of a DecidePart method.

You can change it as you like, e.g. looking down the line and deciding which part gets priority, as long as you choose a part from the WaitingParts list. In assembly stations or machines the DecidePart method searches for each part in the operation plan if all components are available and returns a list of parts to be moved.

OperationPlan

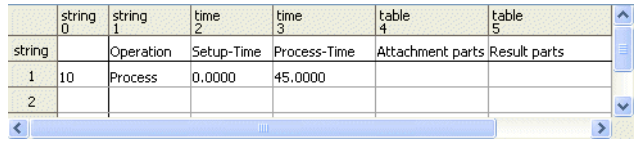
The table OperationPlan defines the behavior of stations and machines. The first level on the operation plan defines which parts are processed on the station or machine. If a part is not in the list it will not be processed on the station.



	string 0	table 1
string	Part	OperationPlan
1	CAR_1	table11
2		
3		

Picture 123:A operation plan with one part

The first column consists of a sub table, where the behavior of the station regarding this special part is defined.



	string 0	string 1	time 2	time 3	table 4	table 5
string		Operation	Setup-Time	Process-Time	Attachment parts	Result parts
1	10	Process	0.0000	45.0000		
2						

Picture 124:The operation plan for part CAR_1

The first two columns are for user defined indexes. Not all columns are used in all stations and machines, e.g. the attachment parts column is only used in assembly and marriage stations and machines.

Operation

A user defined name of this operation.

Setup-Time

The setup time for this part. If there is no entry here, the default setup time will be used.

Process-Time

The process time for this part. If there is no entry here, the default processing time will be used.

Attachment Parts

This column is for assembly stations and machines. It creates a sub table with the name and the quantity of the attachment parts, the first line is always the main part.

Result Parts

This column is for production machines and disassembly stations. It creates a sub tables where all production parts and quantities are defined.

Important Attributes

There are some attributes which are available at almost all objects.

ExitFreeCounter

If the value of the integer attribute ExitFreeCounter is zero, then a part can leave the object. If not, the object is paused or failed.

Objects and Their Special Methods and Tables

Here we go over each object and look after their special methods and abilities defined by tables.

Floorconveyor Objects

Unlike other transport systems, the floorconveyor objects need no load station for a part to enter the system.

How to Use a Floorconveyor Object as a Successor

To define a floorconveyor as a successor of another object, enter the segment in the successor table of the object.

Floorconveyor Basic Segment

The basic segment has only a limited set of tables and methods. It can only transport parts that are announced to it and move them to the successors entered in the successor table. If the successor table is empty and the segment is connected to the next segment, it will move the parts without calling any user methods.

DecidePart

Every time a part wants to enter the segment and it is possible for a part to enter the segment, this method is called. It loops through the WaitingParts list and searches for a part which is on a location where it can be removed, meaning that the location is not paused or failed.

WaitingParts

This is a table where each predecessor enters the parts it want to move to the segment.

	object 0	object 1	boolean 2	string 3
string	Waiting Part	Location	Direction normal	Kind of movement
1	.ApplicationObjects.FactoryCAD	.SDXModels.Lift_Part2.SubSegS	false	Smooth
2				

Picture 125:A sample waiting parts table

WaitingPart

A link to the part

Location

A link to the location the part is on at the moment.

Direction Normal

If the transport direction is backwards (true) or normal (false). (Not used at the moment)

Kind of Movement

“Smooth” or “Place”. Smooth means that the part is on another conveyor, place means it is on an other system.

DecideSuccessor

The method tests if the part type is in the successor list, if not it stops the simulation and reports an error. Else it creates a random number, and compares them to the probability values in the corresponding successor table. Then it returns the chosen successor to the system.

SuccessorTable

The successor table of a floorconveyor basic segment is a basic successor table as described above.

PartsProcessed

This is a table were normally only one row is filled. When a part reaches the end of the segment, it is entered here to prevent that it is announced to different successor when failures or pauses occur.

Floorconveyor Stations

The stations on a floorconveyor consist of a length of conveyor space equal to the biggest part which moves on the conveyor. When a part reaches the end of a station the station behavior is triggered.

Common Methods of Floorconveyor Stations

The stations have some user defined methods in common, which are called at the beginning and the end of processing, when a part is moved from the station and when a part is scrapped on this station.

ProcessStart

Is called when the station starts to process a part. The parameter is the part as an object-link.

ProcessEnd

Is called when the station finished processing a part. The parameter is the part as an object-link.

PartLeft

Is called when the part left the station. The parameter is the part as an object-link.

PartScrapped

Is called when a part is scrapped on the station. The parameter is the part as an object-link.

Common Tables of Floorconveyor Stations

All floorconveyor stations have two tables in common.

PartsProcessed

A part is entered in the PartsProcessed when the act of processing is finished and the station tried to move the part on. In the first column the system enters the part, in the second column is used to store the designated successor.

ScrapTable

This table stores all scrapped parts. Only the type and the quantity of scrapped parts is stored.

Types of Stations

There are different kind of stations which have all different abilities:

Load Station

The load station is used to introduce parts to a floorconveyor system. It can load parts on containers if it is defined in the successor table. The part is moved onto the station, then waits on the station for the processing. After the processing time, it is moved on.

SuccessorTable of a Load Station

In floorconveyor stations has the successor table an additional column. It specifies in the case of a load station which kind of container the part use on this floorconveyor system.

	string 0	table 1	string 2
string	Part	Sucessor data	Used Container
1	TypeA	table11	Admin.ContainerMaster
2	TypeB	table12	Admin.ContainerMaster
3	TypeC	table13	Admin.ContainerMaster
4	BagCase	table14	
5			

Picture 126:Three parts with containers, one without (regard column no. 2)

There are three possible values for the third column:

- Empty

  This means that no container is used for this part on this floorconveyor

- A container master object

  A container master object observes the container managed by it. It has a maximum number of available containers. Each load station will create a needed new container if it is available, or wait for an available container. Container get available if they are deleted by a unload station.

- A container type

  There must be a source of containers for this value. Parts will only be moved on the floorconveyor if a container of the corresponding type is present.

Unload Station

The unload station has the ability to remove parts from a floorconveyor system. It can unload parts from a container if it is defined in the successor table. The part moves on the station, then it is processed, after processing the stations tries to move the part to its successor.

	object 0	integer 1	string 2	real 3	real 4	boolean 5
string	Successor	Probability	Kind of Movement	Movement quantit	Scrap Rate	Use Container
1	.SDXModels.eigen.D	100	Place	1.00	0.00	false
2						

Picture 127:Unload from container to the drain

ProcessStop Station

The process stop station is a machining station. It stops the part, and depending if it is a rigid conveyor or an accumulating one the whole conveyor. When it is done processing the part the station starts the movement anew.

ProcessRun Station

The process run station is the other main type of a machining station. It starts the processing on the part as soon as the parts enters the station, and finishes later on while the part is still moving.

Decision Station and Idle Station

These are subtypes of the ProcessRun station with the same behavior.

Assembly Station

The assembly station is a special type of station, depending on the operation plan it starts processing the main part on the conveyor if all components are available. It then moves all components to the station and deletes them while working on the main part. The assembly station is a subtype of a ProcessRun station.

	string 0	string 1	time 2	time 3	table 4	table 5
string		Operation	Setup-Time	Process-Time	Attachment parts	Result parts
1	10	Assembly			table41	
2						

Picture 128:An operation plan with attachment parts

A sub table of the operation plan of a part is the attachment parts table. It defines which parts are needed in which quantity to assembly the main part.

	string 0	integer 1
string	Part	Quantity
1	Part1	1
2	Part2	1
3		

Picture 129:The attachment parts table

The first part is always the main part, which will leave the station, the next parts are attachment parts, which will be deleted during the assembly process.

Marriage Station

The marriage station has the same behavior as the Assembly station, except it is a subtype of a ProcessStop station, meaning the movement of the part will be stopped until the assembly process is done.

Crosstransfer Floorconveyor Objects

The floorconveyor crosstransfer transports material sideways to their normal direction of transportation. It can only be entered on the stations, similar to the closed transport systems below.

As only parts in the right direction ("up" or "down") can enter the crosstransfer objects, there is a global DecidePart method for the whole crosstransfer.

How to Use a Crosstransfer Floorconveyor Object as a Successor

To define a crosstransfer floorconveyor as a successor of another object, enter the station in the successor table of the object.

Global DecidePart

This method is located at the control object of the crosstransfer and decides which part will enter the crosstransfer based on the global WaitingParts list.

Methods on the Stations

The stations of the crosstransfer decide to which successor the part will be transferred, internally to the next station or externally to another floorconveyor. Therefore they all have the DecideSuccessor method and the SuccessorTable table. For convenience, the WaitingParts table of the station is also accessible.

DecideSuccessor

This method decides based on the successor table to which successor the part will be transferred.

SuccessorTable

Like the normal successor table all successors for a specific part are defined. To move a part along the crosstransfer, the successor is the next station, to move from the crosstransfer the successor is outside the crosstransfer system.

Closed Transport System Objects

Closed transport systems are regular monorail, power and free monorail, electrified monorail and aisles with transporters. What defines a closed transport system is that only at specific points (load and unload stations) material is allowed to enter or leave the system. Only on a load station material can enter a closed transport system, and only on an unload station material can leave the system. On each station the successor station is defined, and only the unload station can have successors outside the system.

How to Use a Closed Transport System Object as a Successor

To define a closed transport system as a successor of another object, enter the station in the successor table of the object.

Stations on a Closed Transport System

A transporter or carrier can get new targets only on stations. Therefore the method DecideSuccessor is only available at stations. The method DecidePart is only available at load stations, as only here a part can enter the system.

A station consists of a length of rail or track equal to the length of a transporter or carrier.

Common Methods of Stations

The stations have some user defined methods in common, which are called at the beginning and the end of processing, when a part is scrapped on this station and which station is the next target.

DecideSuccessor

The method tests if the part type is in the successor list, if not it stops the simulation and reports an error. Else it creates a random number, and compares them to the probability values in the corresponding successor table. Then it returns the chosen successor to the system.

In this method it is possible to change the mode how the successor is decided.

ProcessStart

Is called when the station starts to process a part. The parameter is the part as an object-link.

ProcessEnd

Is called when the station finished processing a part. The parameter is the part as an object-link.

PartScrapped

Is called when a part is scrapped on the station. The parameter is the part as an object-link.

Common Tables of Stations

All stations share some tables, whereas some have more.

SuccessorTable

The successor table of a station is a basic successor table as described above, with the exception that the successor is the next station in the system.

ScrapTable

This table stores all scrapped parts. Only the type and the quantity of scrapped parts is stored.

OperationPlan

The operation plan of a station is a standard operation plan.

Types of Stations

There are different kind of stations which have all different abilities:

Load Station

The load station is the only station in a closed transport system which can introduce material in the system, therefore it has the DecidePart method and the WaitingParts table.

DecidePart

Every time a part wants to enter the system it announces its presence to the load station, and the next time an empty transporter moves on the station, the station stops the transporter and calls this method to decide which part to move.

WaitingParts

In this table each part that wants to enter a closed transport system is noted down. The method DecidePart looks in this table to decide which parts can enter.

UnloadStation

If a part has to leave a closed transport system, it has to do this on an unload station. The successors in the successor table of an unload station have to be outside the closed transport system.

	object 0	integer 1	string 2	integer 3	real 4
string	Successor	Probability	Kind of Movement	Movement quantity	Scrap Rate
1	.SDXModels.Demo_200	100	Place	1	0.00
2					

Picture 130:A successor table of an unload station

ProcessStop Station

The process stop station is a machining station. It stops the transporter or carrier and when it is done processing the part the station starts the transportation anew.

ProcessRun Station

The process run station is the other main type of a machining station. It starts the processing on the part as soon as the transporter or carrier enters the station, and finishes later on while the part is still moving.

Decision Station and Idle Station

These are subtypes of the ProcessRun station with the same behavior.

Assembly Station

The assembly station is a special type of station, depending on the operation plan it starts processing the main part on the transporter or carrier if all components are available. It then moves all components to the station and deletes them while working on the main part. The assembly station is a subtype of a ProcessRun station.

	string 0	string 1	time 2	time 3	table 4	table 5
string		Operation	Setup-Time	Process-Time	Attachment parts	Result parts
1	10	Assembly			table41	
2						

Picture 131:An operation plan with attachment parts

A sub table of the operation plan of a part is the attachment parts table. It defines which parts are needed in which quantity to assembly the main part.

	string 0	integer 1
string	Part	Quantity
1	Part1	1
2	Part2	1
3		

Picture 132:The attachment parts table

The first part is always the main part, which will leave the station, the next parts are attachment parts, which will be deleted during the assembly process. All attachment parts must be outside the closed transport system, only the main part must be in the system.

Additional Table: WaitingParts at the Station

In the table WaitingParts on a station attachment parts waiting to be processed are noted down.

Marriage Station

The marriage station has the same behavior as the Assembly station, except it is a subtype of a ProcessStop station, meaning the movement of the transporter or carrier will be stopped until the assembly process is done.

Buffer Objects

There are three different kind of buffers in FactoryCAD, which differ in the sorting of in- and output. All have some methods and tables in common.

How to Use a Buffer as a Successor

To define a buffer as a successor of another object, enter the buffer in the successor table of the object.

Common Methods in Buffers

Buffers have all custom methods which define the behavior of the object, and all trigger methods like when a part starts processing or leaves the object.

DecidePart

This is the standard method for all objects. It decides which part of the waiting parts list will enter the buffer.

DecideSuccessor

The decide successor method is also the standard method for all objects, which will decide depending on the successor list to which successor the part will be transferred.

ProcessStart

Is called when the station starts to process a part. The parameter is the part as an object-link.

ProcessEnd

Is called when the station finished processing a part. The parameter is the part as an object-link.

PartLeft

Is called when the part left the station. The parameter is the part as an object-link.

PartScrapped

Is called when a part is scrapped on the station. The parameter is the part as an object-link.

Special Methods in Buffers

Only the RestoreSequence buffer has two special methods which allows to define which attribute is the one defining the sequence.

InsertSortCriteria

This method is called every time a part is ready to leave the RestoreSequence buffer and allows the user to define which criteria is the sorting criteria which is entered in column three of the parts processed table.

SortParts

Is called every time after the InsertSortCriteria method and sorts the sequence of parts in the buffer.

Common Tables in Buffers

All buffers share four tables: WaitingParts, SuccessorTable, PartsProcessed and ScrapTable.

WaitingParts

In this table each part that wants to enter the buffer is noted down. The method DecidePart looks in this table to decide which parts can enter.

SuccessorTable

The successor table of a buffer is a basic successor table as described above.

PartsProcessed

A part is entered in the PartsProcessed table when the act of processing is finished and the buffer tried to move the part on. In the first column the system enters the part, in the second column is used to store the designated successor. In the RestoreSequence buffer the third column is used to store the sorting criteria.

ScrapTable

This table stores all scrapped parts. Only the type and the quantity of scrapped parts is stored.

Machine Objects

Machines are standard processing units, so they have all the basic behavior of objects. Some machines use information differently as others, which we will describe here.

How to Use a Machine as a Successor

To define a machine as a successor of another object, enter the machine in the successor table of the object.

Common Methods in Machines

Machines have all custom methods which define the behavior of the object, and all trigger methods like when a part starts processing or leaves the object.

DecidePart

This is the standard method for all objects. It decides which part of the waiting parts list will enter the machines.

DecideSuccessor

The decide successor method is also the standard method for all objects, which will decide depending on the successor list to which successor the part will be transferred.

ProcessStart

Is called when the station starts to process a part. The parameter is the part as an object-link.

ProcessEnd

Is called when the station finished processing a part. The parameter is the part as an object-link.

PartLeft

Is called when the part left the station. The parameter is the part as an object-link.

PartScrapped

Is called when a part is scrapped on the station. The parameter is the part as an object-link.

Common Tables in Machines

Machines have all the standard tables and additionally the operation plan table.

WaitingParts

In this table each part that wants to enter the machine is noted down. The method DecidePart looks in this table to decide which parts can enter.

SuccessorTable

The successor table of a machine is a basic successor table as described above.

PartsProcessed

A part is entered in the PartsProcessed table when the act of processing is finished and the machine tried to move the part on. In the first column the system enters the part, in the second column is used to store the designated successor.

ScrapTable

This table stores all scrapped parts. Only the type and the quantity of scrapped parts is stored.

OperationPlan

The operation plan of a station is a standard operation plan.

Types of Machines

All types of machines have different abilities. Most of the abilities are defined in the operation plan table.

SingleOperation

The single operation machine is the basic one part a time machine.

BatchOperation

More than one part must enter the batch operation machine to start the processing. How many is defined by the parts per cycle setting.

MultiStation

This machine consists of many independent stations bundled together. They share the exit and entry behavior, but process independently from each other.

Assembly

The assembly machine assembles parts as defined in the operation plan, in the subtable AttachmentParts.

	string 0	string 1	time 2	time 3	table 4	table 5
string		Operation	Setup-Time	Process-Time	Attachment parts	Result parts
1	10	Assembly			table41	
2						

Picture 133:A operation table with attachment parts

The first entry in the attachment parts table is the main part, all other parts are attachment parts and will be deleted during the process.

	string 0	integer 1
string	Part	Quantity
1	Part1	1
2	Part2	1
3		

Picture 134:Part1 is the main part, Part2 only a attachment part

It is also possible to assemble more than one part a time, which is defined in the part per cycle setting. The assembly machine has a special DecidePart method, which return not only a part but a list of parts which is a bill of material for an assembly process.

```
:table[object,object,boolean]
is
    i,j,k,l      : integer;
    RowNo        : integer;
    EntityType   : string;
    Quantity     : integer;
    Found        : integer;
    Available    : boolean;
    PartFound    : boolean;
do
    result.create;
    i := 1;
    PartFound := false;
    while i <=  WaitingParts.ydim and not PartFound loop
        RowNo := OperationPlan.getRowNo(WaitingParts[0,i].EntityType);
        if RowNo = -1 and WaitingParts[0,i].getAttrNo("Variant") /= 0
        then
            RowNo := OperationPlan.getRowNo(WaitingParts[0,i].Variant);
        end;

        if RowNo > 0
        then
```

Picture 135:The DecidePart method of an assembly machine

Note: The first column of the result table is a part, the second one the location of the part, the third column defines if the part is part of a batch on a buffer. Only the last part of a batch on a buffer should have `true` here.

Production

The production machine disassembles parts and uses the result part subtable for information. In this subtable there are entries which part and which quantity of parts is produced of one entry part.

Workcenter Objects

Workcenter are basic processing units, so they have only the basic behavior of objects.

How to Use a Workcenter as a Successor

To define a workcenter as a successor of another object, enter the workcenter in the successor table of the object.

Common Methods in Workcenters

Workcenters have only the basic two methods to define the entry and exit strategies.

DecidePart

This is the standard method for all objects. It decides which part of the waiting parts list will enter the workcenter.

DecideSuccessor

The decide successor method is also the standard method for all objects, which will decide depending on the successor list to which successor the part will be transferred.

Common Tables in Workcenters

Workcenters have all the standard tables and the operation plan table.

WaitingParts

In this table each part that wants to enter the workcenter is noted down. The method DecidePart looks in this table to decide which parts can enter.

SuccessorTable

The successor table of a workcenter is a basic successor table as described above.

PartsProcessed

A part is entered in the PartsProcessed table when the act of processing is finished and the workcenter tried to move the part on. In the first column the system enters the part, in the second column is used to store the designated successor.

ScrapTable

This table stores all scrapped parts. Only the type and the quantity of scrapped parts is stored.

OperationPlan

The operation plan of a workcenter is a standard operation plan.

Types of Workcenters

Only the assembly subtype of the workcenter has a special behavior.

Assembly

The assembly workcenter has the same ability as the Assembly machine without the possibility to process more than one part a time.

Dock Objects

The docks are entrance and exits of the facility, so they have only basic abilities.

How to Use a Dock as a Successor

To define a dock as a successor of another object, enter the dock in the successor table of the object.

Common Methods in Docks

Dock have only the basic two methods to define the entry and exit strategies.

DecidePart

This is the standard method for all objects. It decides which part of the waiting parts list will enter the dock.

DecideSuccessor

The decide successor method is also the standard method for all objects, which will decide depending on the successor list to which successor the part will be transferred.

Common Tables in Docks

Dock have all the standard tables and the operation plan table.

WaitingParts

In this table each part that wants to enter the dock is noted down. The method DecidePart looks in this table to decide which parts can enter.

SuccessorTable

The successor table of a dock is a basic successor table as described above.

PartsProcessed

A part is entered in the PartsProcessed table when the act of processing is finished and the dock tried to move the part on. In the first column the system enters the part, in the second column is used to store the designated successor.

ScrapTable

This table stores all scrapped parts. Only the type and the quantity of scrapped parts is stored.

OperationPlan

The operation plan of a dock is a standard operation plan.

Robot Objects

The robot objects have the same abilities as machines except there are only three basic types: the normal robot, the material handling robot and the assembly robot. Both the normal robot and the material handling robot derive from the single operation machine, the assembly robot derives from the assembly machine.

How to Use a Robot as a Successor

To define a robot as a successor of another object, enter the robot in the successor table of the object.

Index

About Siemens PLM Software

Siemens PLM Software, a division of Siemens Automation and Drives (A&D), is a leading global provider of product lifecycle management (PLM) software and services with 4.6 million licensed seats and 51,000 customers worldwide. Headquartered in Plano, Texas, Siemens PLM Software's open enterprise solutions enable a world where organizations and their partners collaborate through Global Innovation Networks to deliver world-class products and services. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.

SIEMENS

Division headquarters

United States
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
972 987 3000
Fax 972 987 3398

Regions

Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
800 498 5351
Fax 972 987 3398

Europe
Norwich House Knoll Road
Camberley, Surrey
GU15 3SY
United Kingdom
44 1276 702000
Fax 44 1276 705150

Asia-Pacific
Suites 6804-8, 68/F., Central Plaza
18 Harbour Road, WanChai
Hong Kong
852 2230 3333
Fax 852 2230 3210

10/07