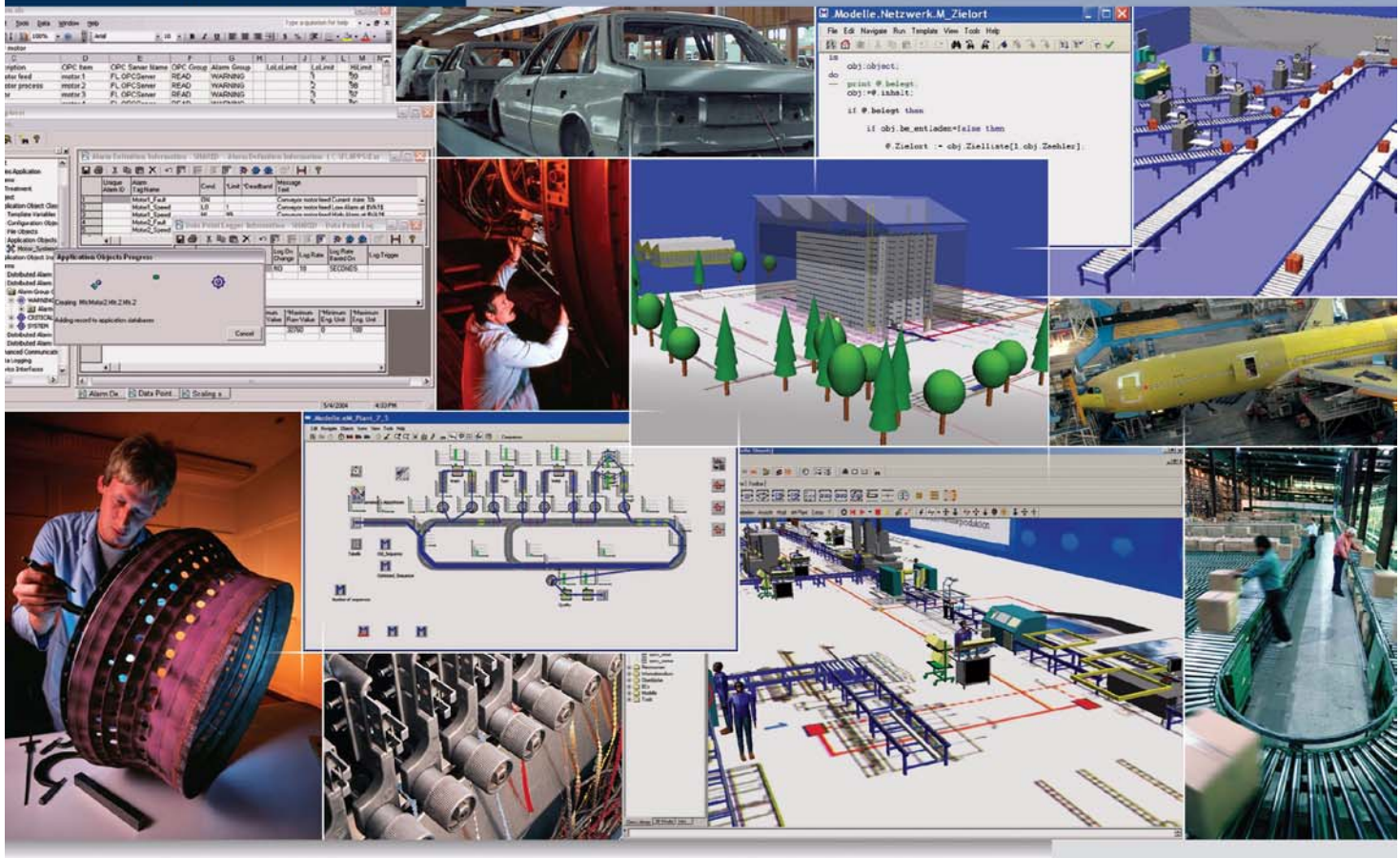
Plant Simulation Shop library

Reference manual

Siemens PLM Software

www.siemens.com/plm



TECNOMATIX

SIEMENS

Proprietary and Restricted Rights Notice

TECNOMATIX
Plant Simulation

SIEMENS

Plant Simulation

Shop Library

Version 9.0

December 2008

Table of Contents

1

Plant Simulation Shop

Plant Simulation Shop is a library for modeling, simulating, animating, and evaluating discrete production systems in the producing industries. Plant Simulation Shop is based on Plant Simulation and requires an Plant Simulation Development License or an Plant Simulation Application License. Using the library requires knowledge in Plant Simulation.

Note: We intend to cover your respective modeling task as completely as possible in the library. Based on the flexibility and the universality of the AOL, the objects and their relations to each other are oftentimes complex. Customizing the AOL requires a thorough understanding of the functionality and is thus not always an easy task. For this reason we recommend, before using the AOL, to check if the built-in functionalities meet your modeling needs. If you have to customize the AOL, we recommend to carefully analyze the cost-benefit ratio. The AOL is best suited for quickly creating prototypes and as a reference tool for libraries you create.

Simulation, Animation and Evaluation

Use the Plant Simulation Shop objects to graphically and interactively create the production structure. Order planning is based on the material requirements calculation during the forecast phase. For controlling the material flow and the order flow, we provide order priorities and common rules for entering and processing orders. You can also quickly and easily create additional rules. By watching the simulation and the animation, you can check and optimize the layout, the dimensioning and the control strategies of the production system. Production orders are based on this, on the bills of material and on the work schedules. The program shows the results of the simulation, such as the utilization of resources, costs, etc. in dialogs, and tables. It shows the production plans as Gantt charts.

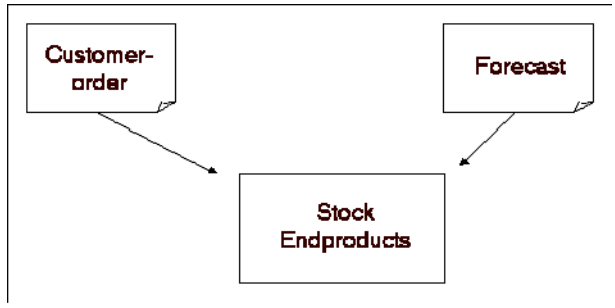
Plant Simulation supports the engineer in planning the production system and the scheduler and the methods engineer in optimizing the production plan during the operative business.

For this reason you can also use Plant Simulation Shop to complement an existing PPS system to:

- Optimize the PPS system parameters.
- Check the feasibility of production plans under realistic conditions and to evaluate different scenarios.
- Create detailed plans of given order profiles based on current data.

Material Requirements Planning Basics

Material requirements planning serves for better accommodating the forecast sales. Using these numbers, the planner then creates a production plan. Customer orders arriving late are charged against the planned orders from the forecast. When more product is needed, rush orders are started.



Material requirements planning basics

The program then creates the planned orders using the data of the forecast. Creating the production order, by taking the lot size rules for the individual products into account, and by bills explosion and processing the components, results into this planning sequence:

1. Add the planning data for the main products (*PlanOrder*).
2. Create the production orders for the main products (*ProdOrder*).
3. Create the derived planning orders for the components (*DerivedOrder*).
4. Create the production orders for the components (*ProdOrder*).
5. Repeat step 3 and step 4 until all products are taken care of.

Note that we only look at an unloaded system, i.e. we do not calculate any capacities. For this, the planner has to manually plan safety margins.

Modeling

Define the parts to be produced in a forecast. You find the tables *MasterData*, *Forecast* and *ProductMix* in the object *MRP* (Material Resource Planning) within the folder *logistic*. Enter the order planning into the tables *Forecast* and *ProductMix*, enter the parts that are relevant to the simulation into the table *MasterData*. The parts are moved to the machines and the work stations according to the work schedule. For this we provide the production control objects *ProductionManager* and the *Station* and the *Stationpool* in the folder *production*. You can connect the stations with a material flow system. This allows you to model a great variety of production facilities. As a rule the individual stations will have different capacities and process parts with different speeds. This results in different cycle times, which you will have to compensate for by using buffers, etc. The *ProductionManager* enables you to move parts from the buffer to the next station using the rules you selected in the dialogs. Another characteristic of most production facilities is the possibility to have several stations execute a processing step. The *ProductionManager* selects an available station according to different rules.

Plant Simulation Shop Objects

Plant Simulation Shop provides these objects:

MRP

- Master data of all existing parts (procurement type, work schedules including operations, processing and set-up times, bills of material, in-house production times, delivery times).
- Forecast
- ProductMix
- ProductionPlan

ProductionManager

- Manages production orders (batches to be processed).
- Manages the processing stations and their operations.
- Provides rules for controlling and processing parts and orders:
  - Entry Queueing
  - Process Sequencing
  - Resource Selection

WarehouseManager

- Manages the different warehouses.
- Checks availabilities.
- Manages stock shortfalls and releases waiting orders.

Warehouse

- Manages the warehouse (to-bin transfer, inventory).
- Checks the availability of the stored materials.

PurchasingManager

- Manages orders (supplier, average cost per unit, delivery times).
- Issues purchasing orders.

CustomerOrder

- Manages customer orders (Order number, product, quantity, order and scheduled deliverty time, through the simulation the delivery times).
- Usage of the forecast table, modifing it.

Station, StationPool

- Name
- (StationPool) number of stations
- Operations to be executed, production costs
- Processing and set-up times, failure times
- Evaluations, costing
- Simulates failures
- Add a buffer

Analyzer

- Lists late orders
- Costs with cost center and cost groups split up by: Order costs, set-up costs, and processing costs
- Statistics of the individual stations: Utilization, set-up times and processing times
- Throughput times

StockEvaluation

- Shows the circulating stock and the total stock.
- Total or interval-related
- Product-specific

Gantt

- Start date
- Shows data as graphics or in lists.
- Start time and end time of the operations.
- Shows the utilization of resources.

Before looking at the detailed description of the objects, let us view a first example. The demo model *demo* is located in the folder *.models*.

2

Example

A metal processing plant produces a transmission with shafts and toothed gears. The products have different work schedules and different set-up times and processing times. The plant provides these resources: Two lathes, a milling machine, a drill, and a shipping department. The forecast is to cover three months.

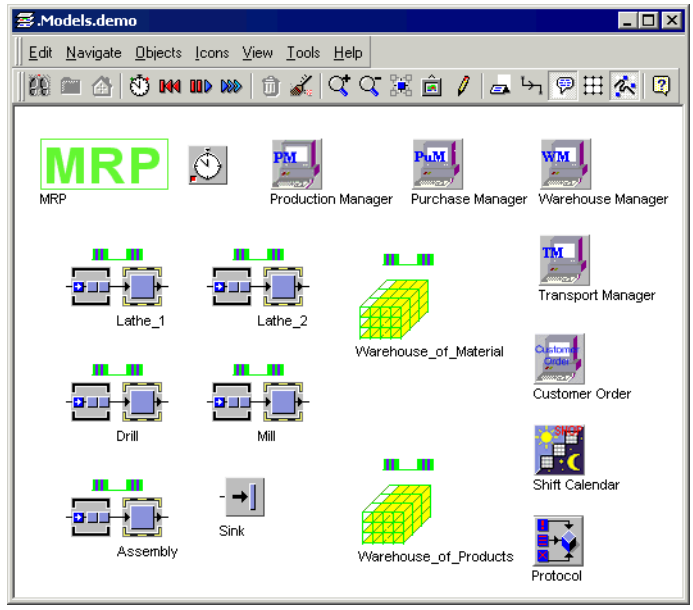
For this production system we want to employ simulation to arrive at the optimal production strategy. We want answers for these questions:

- Which order sequences result for the production order control and for machine scheduling when using different rules?
- Which utilization results for the individual machines?
- Which costs accrue for the individual machines and groups of machines?
- Which set-up times, processing times and throughput times do the machines have?
- Are any orders late?

To model the production, first open a *Frame*. Then insert the object *MRP*, a single *ProductionManager* object, 5 *Station* objects with a *Buffer* object and a *DockingStation* each, a *WarehouseManager* with two *Warehouses,* a *PurchaseManager* and a *TransportManager*.

If you purchased Plant Simulation Gantt, you can also graphically display machine occupation plans. For this you have to insert the object *Gantt*. For evaluating costs, you have to insert the objects *Analyzer* and the object *StockEvaluation*.

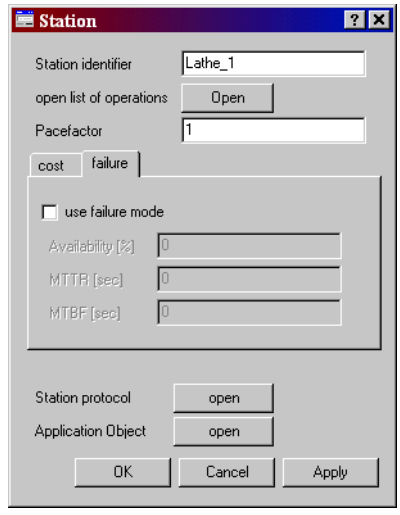
For querying information about transport times, you have to insert the object *TransportManager* and the object *DockingStation* at each processing station, which is called *Station*. The screenshot below shows a possible layout of the model.



The simulation model depicting the production

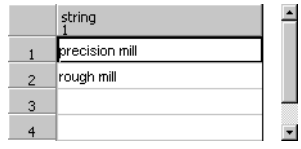
Station

The processing station is represented by the object *Station*. Double-click the icon of the object to open its dialog.



Dialog of the *Station*

Enter the name of the station. To enter the operations into the station, click the button Open.



The list of operations of the *Station*

For cost evaluation you have to activate cost gathering and enter the cost center. Enter the number of stations of the entry buffer into the associated object *Buffer*.

If you would like to also simulate failures, select the check box Use failure mode. Then you can enter values for the availability and MTBF and MTTR to define the failures.

Materials Requirement Planning

Create the orders for the production system in the object *MRP*. Enter the master data and the forecast data into the dialog. Double-click the icon of the object, which you inserted into the *Frame*, to open its dialog.

Forecast Data

Click the button Edit Forecast to open the list of the forecast data, which may look like this:

	string 0	integer 1	integer 2	integer 3	integ 4
string	productGroup	1.8.2002	1.9.2002	1.10.2002	
1	Products	30	35	28	
2	Spare parts	15	17	14	
3					

Forecast data in material requirements planning

Enter the amount of products into the individual product groups and the intervals.

Product Mix Table

Click the button Edit Product Mix to open the table that lists the product groups of the individual products according to the table above:

	string 1	table 2
string	ProductGroup	products
1	Products	products
2	Spare parts	products
3		

	string 1	real 2
string	productId	percentage
1	1300-100	60.00
2	1300-200	40.00
3		
4		

The general product mix table an a product mix subtable

Enter the amount of products into the individual product groups and the intervals.

MRP Master Data

Click the button Edit Master Data to open the list of the master data of all parts. Enter the procurement type of the part into the second column. The string "I" stands for a part that is produced in-house. The string "E", for external, stands for a part that a supplier provides. Your master data might look like this:

	string 0	string 1	string 2	table 3	table 4	integer 5	integer 6	integer 7	table 8	time 9	table 10
string	productId	product	PT	BOM	OperationPlan	inhouseProdTime	calcProdTime	calcSetupTime	LotSizeData	deliveryTime	warehouse
1	1300-100	transmission, 1	I	bom	op	6000	6000	9600	lot		warehouse
2	1300-200	transmission, 2	I	bom	op	6000	6000	9600	lot		warehouse
3	2100-100	shaft, 100 mm	I	bom	op	3300	3300	27600	lot		warehouse
4	2100-200	shaft, 200 mm	I	bom	op	5220	5220	23400	lot		warehouse
5	7100-000	transmission housing	E			0			lot	8:00:00:00.000	warehouse
6	7300-060	round bar, 60 mm	E			0			lot	8:00:00:00.000	warehouse
7	7400-050	gearwheel, 50 tooths	E			0			lot	8:00:00:00.000	warehouse
8											

Master data of all parts

Work Schedules within the Master Data

Double-click the right part of a cell of the column *OperationPlan* to open the work schedule of the corresponding part:

	integer 1	string 2	time 3	time 4	time 5	real 6
string	operationNo	operation	setupTime	processingTime	personalTime	base
1	10	rough turn	2:00:00.0000	12:00.0000	0.0000	1.00
2	20	precision turn	2:00:00.0000	16:00.0000	0.0000	1.00
3	30	rough mill	2:40:00.0000	12:00.0000	0.0000	1.00
4	40	drill	1:00:00.0000	15:00.0000	0.0000	1.00
5						

Work schedule of the part *shaft, 100 mm*

A part supplied by a supplier does not require a work schedule. Enter the sequence of operations with the respective in-house productions shares of the processing and set-up time.

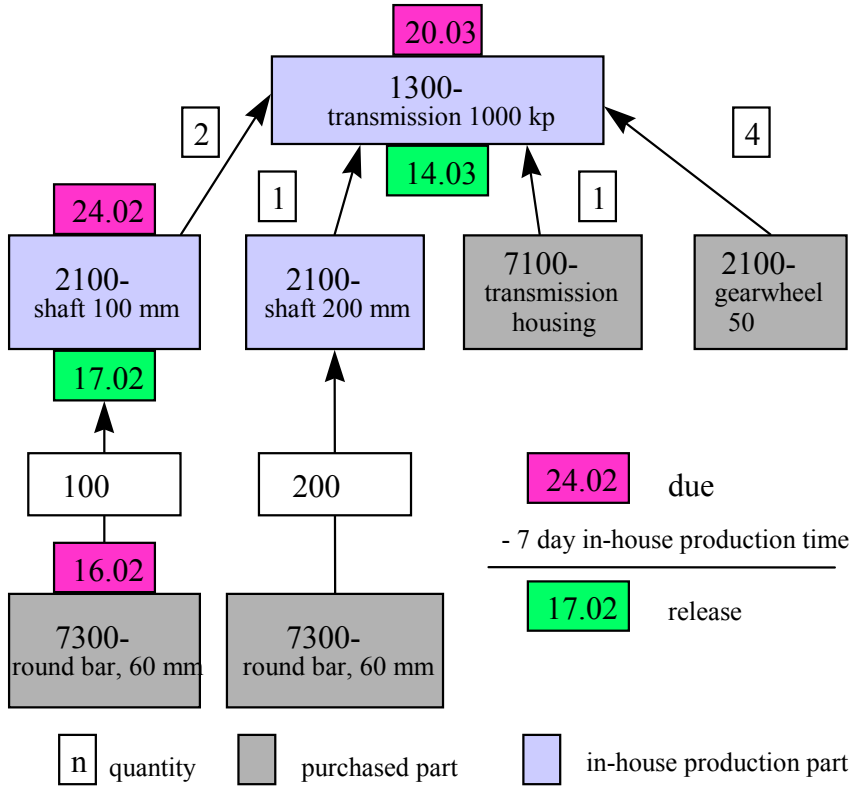
Bills of Material within the Master Data

An in-house production part itself may be composed of several parts. For this reason you can define a bill of material for these parts. Enter it into the column *BillOfMaterial*.

	integer 1	string 2	integer 3	string 4
string	positionNo	partId	factor	unit
1	10	2100-100	4	PC
2	20	2100-200	2	PC
3	30	7100-000	1	PC
4	40	7400-050	8	PC
5				

Section of the bill of material masterDataL[3,2] of the part 1300-200

The following figure shows the bill of materials tree of the part 1300-100. It also shows the scheduling of a customer order with the due date March 20.



Scheduling a production order for the part 1300-200

ProductionManager

The object *ProductionManager* controls the entire production process.

Dialog of the *ProductionManager*

Production Orders within the ProductionManager

Click the button Orders to open the list of the production orders, which may look like this:

	string 0	string 1	string 2	table 3	table 4	integer 5	datetime 6	datetime 7
string	orderNo	EntityType	EntityName	OperationPlanL	billOfMaterial	Priority	releaseDate	dueDate
1	PO-2007	2100-100	shaft, 100 mm	table31	table41	1	31.07.2002 13:40:00.0000	31.07.2002 21:20:00.0000
2	PO-2008	2100-100	shaft, 100 mm	table32	table42	1	31.07.2002 13:40:00.0000	31.07.2002 21:20:00.0000
3	PO-2016	2100-200	shaft, 200 mm	table33	table43	1	31.07.2002 14:50:00.0000	31.07.2002 21:20:00.0000
4	PO-2017	2100-200	shaft, 200 mm	table34	table44	1	31.07.2002 14:50:00.0000	31.07.2002 21:20:00.0000
5	PO-2009	2100-100	shaft, 100 mm	table35	table45	1	31.07.2002 16:20:00.0000	01.08.2002 00:00:00.0000
6	PO-2018	2100-200	shaft, 200 mm	table36	table46	1	31.07.2002 17:30:00.0000	01.08.2002 00:00:00.0000
7	PO-2001	1300-100	transmission, 1	table37	table47	1	31.07.2002 21:20:00.0000	01.08.2002 00:00:00.0000
8	PO-2004	1300-200	transmission, 2	table38	table48	1	31.07.2002 21:20:00.0000	01.08.2002 00:00:00.0000

A section of the production orders list of the *ProductionManager*

The List of Operations within the ProductionManager

Click the button Show in the dialog of the *ProductionManager* to open a list of all operations of the different stations.

	string 0	list 1	list 2
string	Operation	StationL	PartL
1	rough turn	list11	list21
2	precision turn	list12	list22
3	rough mill	list13	list23
4	precision mill	list14	list24
5	drill	list15	list25
6	assembly	list16	list26
7			

List of all operations

For each operation the program creates a table that contains all stations that execute this operation and a table of all parts waiting for this operation.

Plant Simulation Shop creates this list automatically during Reset.

List of Stations within the ProductionManager

This list contains the same items as the list of operations. Instead of being sorted by operations it is sorted by stations.

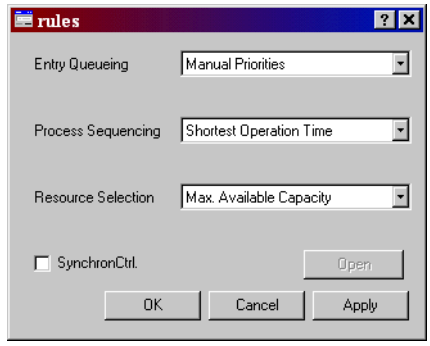
	object 0	string 1	string 2	string 3
string	work station	operation	operation	
1	.Models.demo.Lathe_1	rough turn		
2	.Models.demo.Lathe_2	precision turn		
3	.Models.demo.Mill	rough mill	precision mill	
4	.Models.demo.Drill	drill		
5	.Models.demo.Assembly	assembly		
6				
7				

List of all stations

When you change the operations in this list, the program propagates the changes back to the stations.

Selecting Rules in the ProductionManager

Click the button Select in the dialog of the *ProductionManager* to open the dialog Rules. Select the rules for dispatching orders and machines here.



Select dispatch rules

WarehouseManager

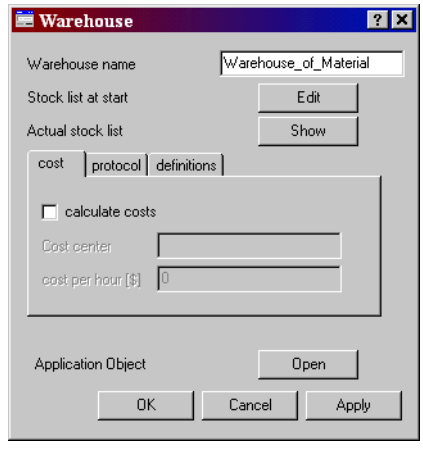
Double-click the icon of the *WarehouseManager* to open its dialog window.

- Click the button Show to open a list of all warehouses inserted into your model. Before starting a production order the program checks, if the amount of items contained in the bill of materials is available in the warehouses.
- Click the button Show to show all currently missing purchase parts.

Dialog of the *WarehouseManager*

Warehouse

To open the dialog of the *Warehouse*, either double-click its icon or double-click the right part of a cell of the list of the warehouses.



Dialog of the *Warehouse*

Initial Inventory

To open a table, into which you enter the initial inventory at the beginning of the simulation, click the button Edit.

Current Inventory

Click the button Show to show the current inventory or the inventory at the end of the simulation.

Additional Information for the Warehouse

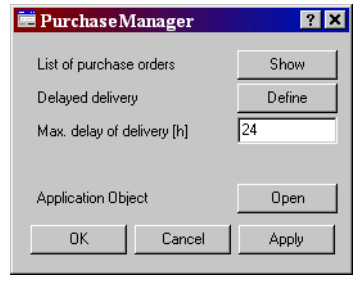
If you also want to analyze costs, you have to activate cost gathering and enter the expense factor and the cost center into the dialog.

Click the button Entrance and enter the number of stations of the entrance buffer.

Click the button Exit and enter the number of stations of the exit buffer.

PurchasingManager

Double-click the object *PurchasingManager* to open its dialog. Click the button Show to open a table containing the purchase orders.

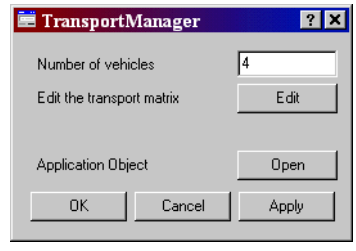


Dialog of the Purchasing Manager

You can also define the portion of delayed deliveries and enter the maximum number of hours the delivery can be late.

TransportManager

Double-click the object *TransportManager* to open its dialog.



Dialog of the *TransportManager*

Enter the number of transporters that fulfill the transport orders. Click the button Edit to open a table containing the transport times. When you reset the model or when you manually start the *Validator*, the program automatically enters the column index and the row index. Otherwise you have to manually enter the paths of the processing stations and the warehouses The values in the table are the time it takes to transport a part from point A to point B.

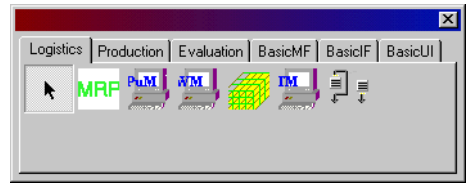
	object 0	time 1	time 2	time 3	time 4	time 5	time 6
object		ParkingPlace	.Models.demo.Lathe_1	.Models.demo.Lathe_2	.Models.demo.Mill	.Models.demo.Dril	.Models.demo
1	ParkingPlace	0.0000	1:00:00.0000	1:00:00.0000	1:00:00.0000	1:00:00.0000	1:00:00.0000
2	.Models.demo.Lathe_1	1:00:00.0000	0.0000	1:00:00.0000	2:00:00.0000	3:00:00.0000	4:00:00.0000
3	.Models.demo.Lathe_2	1:00:00.0000	1:00:00.0000	0.0000	1:00:00.0000	2:00:00.0000	3:00:00.0000
4	.Models.demo.Mill	1:00:00.0000	2:00:00.0000	1:00:00.0000	0.0000	1:00:00.0000	2:00:00.0000
5	.Models.demo.Drill	1:00:00.0000	3:00:00.0000	2:00:00.0000	1:00:00.0000	0.0000	1:00:00.0000
6	.Models.demo.Assembly	1:00:00.0000	4:00:00.0000	3:00:00.0000	2:00:00.0000	1:00:00.0000	0.0000
7	.Models.demo.Warehouse_of_Material	1:00:00.0000	5:00:00.0000	4:00:00.0000	3:00:00.0000	2:00:00.0000	1:00:00.0000

The transport table

Now you can start the simulation.

3

The Plant Simulation Shop Objects



The Plant Simulation Shop Toolbox

The library Plant Simulation Shop provides these objects:

- Material requirements planning (*MRP*)
- Production manager (*ProductionManager*)
- Processing station (*Station*)
- Processing station pool (*Stationpool*)
- Warehouse manager (*WarehouseManager*)
- Warehouse (*Warehouse*)
- Purchasing manager (*PurchaseManager*)
- Customer order manager (*CustomerOrder*)
- Transport manager (*TransportManager*)
- Docking station (*DockingStation*)
- Part or batch (*part* = Production order)
- Analyzer (*Analyzer*)
- Gantt (*Gantt*)
- Stock evaluation (*StockEvaluation*)

Material Requirements Planning MRP

The object *MRP* handles the material requirements planning. You typically create your planning at the beginning of the simulation. Then, the different orders are passed as lots, usually at the beginning of each day, to the objects *Production-Manager* and *PurchaseManager* that fulfill the orders. Enter the master data of the parts, the forecast, and the product mix into the object *MRP*. In addition the tables *ProductionPlan* and *ProductOrderList* are provided. Into the table *ProductionPlan* the program enters the planning data and the production data according to the products. Into the table *ProductOrderList* the program enters the production data sorted by release date.

The register card Master Data

On this registercard are all data concerning the parts.

Master Data of the Parts

Click the button Edit next to Edit master data, to open the table *MasterData*. Enter the part types which are to be processed and to be purchased.

ProductId

We predefined the row index as the *productId*. This is a unique identifier for the object, such as the master data number, etc.

Product

Enter a short description of the product into the column *product*.

PT

Enter the procurement type of the part into the column *PT*:

- "`I`", for internal, if the part is to be produced.
- "`E`", for external, if the part is to be purchased.

BOM

Double-click the right part of a cell of the column *BOM* to open the bill of materials of that part, compare Section of the bill of material masterDataL[3,2] of the part 1300-200 on page 7.

Enter into the bill of materials *BOM*:

- All positions of the bill of materials are numbered, see the column *positionNo*.
- Each component is identified by its *partId*, that corresponds with the *productId* of the *MasterData* list.
- Enter how often the component is required for the assembly of the respective part into the column *factor*.
- Enter the unit for this bill of materials into the column *unit*. This unit has to correspond to the associated unit in the *MasterData* list.

OperationPlan

Double-click the right part of a cell of the column *OperationPlan* to open the work schedule of the corresponding part, compare Work schedule of the part shaft, 100 mm on page 7. The sequence of the processing steps is the same as the sequence within the work schedule. (compare the column *operationNo*).

The part type determines the set-up time and the time for the workers. The present version of the AOL does not use the column *personalTime* yet.

InhouseProdTime

Enter the in-house production time into the column *inhouseProdTime*. It is the basis for the material requirements planning. You can ehter the time the program calculated into the column *calcProdTime*. We recommend to add a safety margin for failures of the system, double-assignments of machines, etc.

CalcProdTime & CalcSetupTime

When you close the master data table, the program enters the production time it calculated into the column *calcProdTime* and the set-up times it calculated into the column *calcSetupTime*. The program calculates these times by adding up the times of the individual steps.

LotSizeData

Double-click the right part of a cell containing an entry in the column *LotSizeData* to open a sub-table. Enter rules for the lot sizes of the product.

	string 1	integer 2	integer 3	table 4	string 5
string	LotSizeRule	LotSize	MaxStock	LotProfil	
1	Fixed	5			

Lot size rules for a product

- LotSizeRule: You can enter:
  - `Fixed` for a fixed lot size, each requirement is split up into fixed lot sizes.
  - `Exact` for an exact lot size, each requirement results in an order of the same size.
  - `Replenish` for replenishing the warehouse to maximum capacity after each requirement.

- `Profile` for which each requirement is translated via a profile table into a lot size.
- Not a lot size rule, but a production control strategy: `Kanban`

- LotSize*:* A fixed lot size, which the rules *Fixed* and *Kanban* require.
- MaxStock: Is the maximum amount of stock the warehouse can accommodate, which the rule *Replenish* requires.
- LotProfil: Is a table that contains the profile for the lot size, which the rule *Profile* requires.

	integer 1	integer 2	integer 3
string	From	Until	LotSize
1	1	2	2
2	3	5	5
3	6	10	10
4	11	100	50
5			

A lot profile for the rule *Profile*

DeliveryTime

Enter the time a purchased part needs to arrive at the warehouse of the production after it has been ordered into the column *deliveryTime* of the master data table.

Warehouse

Double-click the right part of a cell containing an entry in the column *warehouse* to open a sub-table. Enter data for warehousing here.

	string 1	string 2	integer 3	integer 4	time 5
string	Warehouse	Unit of Quantity	SafetyStock	NumKanban	StartZeit f. Kanban
1	Warehouse_of_Products	PC			

Warehouse view

- Warehouse*:* Enter the name of the warehouse that stores the part.
- Unit of Quantity: Enter the unit for the amount used for storing the part.
- SafetyStock*:* The safety stock of the store which the material requirements planning cannot fall short of.
- NumKanban: Is the number of the available Kanban cards or Kanban bins.
- StartTime for Kanban: Enter the time you want to start the Kanban cards or bins.

Costs

Double-click the right part of a cell containing an entry in the column *costs* to open a sub-table. Enter data about the cost view.

	real 1	real 2	integer 3	real 4
string	cost_of_purchase	cost_of_factory	unit_of_cost	production costs
1	50.00	210.00	1	210.00

Cost view

- cost_of_purchase: Purchase price
- cost_of_factory: Production costs for in-house production
- unit_of_cost: most of the time per 1, 10, 100, pieces, etc.
- production costs:

FixedInhouseProductionTime

If you want to use a fixed production time per production order regardsless of the lot size in the planning process, insert this time in this cell.

ShiftCalender

If the product is only processed on stations with a specific shiftplan, insert this shiftcalendar in the cell. This calendar will be used for scheduling.

Loading and Saving

Click the button Save to save the master data table as a text file. Click the button Load to import master data.

Use calculated production time

Triggers the usage of the production time calculated from the operationplan for the planning.

The register card forecast

On this card all funtionality concerning the forecast is given.

The Forecast Table

Click the button Edit to open the table *forecast* of the material forecast calculation. Enter the sales forecast for the individual product groups here.

	string 0	integer 1	integer 2	integer 3	integ 4
string	productGroup	1.8.2002	1.9.2002	1.10.2002	
1	Products	30	35	28	
2	Spare parts	15	17	14	
3					

A sample forecast table

The checkbox UseProductMix

With this checkbox you can control the behaviour of the forecast table. If it is not checked you have to use the productIds in the forecast table, if it is, you have to use product groups defined in the product mix table.

Defining the Product Groups

Click the button Edit below the button for the forecast. Enter the individual products that form the product groups into the table *productmix*. This is a nested table. Enter the product groups into the first column. Enter the names of the subtables into the cells, and enter the names and the percentages of the products into the subtables.

	string 1	table 2
string	ProductGroup	products
1	Products	products
2	Spare parts	products
3		

	string 1	real 2
string	productId	percentage
1	1300-100	60.00
2	1300-200	40.00
3		
4		

Breakdown of the product group products

The register card planning

All funtions and tables concerning the planning.

The Production Plan

Click the button Show next to Show Production Plan to open the table *productionschedule*. It contains the entire planned production sequence, including the planned orders, the production orders and the purchase orders. The program automati-

cally enters the data from the forecast table, etc. here. This schedule is split up into the individual products. If the table is empty, the program evaluates the forecast before opening it and writes the result as planning orders to the table. Creating the production orders then also creates the planned orders for the components.

	productId	Schedule
1	1300-100	table11
2	1300-200	table12
3	2100-100	table13
4	2100-200	table14

	OrderNo	StartDate	EndDate	OrderType	Quantity	Stock
1						0
2			01.08.2002 00:00:00.0000	PlanOrder	18	-18
3			01.09.2002 00:00:00.0000	PlanOrder	21	-39
4			01.10.2002 00:00:00.0000	PlanOrder	16	-55

Sample of a production plan while planning

The checkbox "Setup only the first time in a series"

If this checkbox is not set, the setup time will be considered in every production order, if it is then in a series of production orders originated by a plan or derived order only the first production order will have a setup time included, the rest is considered to be processed on the same machine without the need for setup.

The button "Start creating production orders"

If you want to start planning by yourself, you have to use this button. It starts the planning cycle and fills the production schedule and the other tables.

The Complete Production Order List

The production order list shows the order sorted by time. The columns *released* and *releasedDate* show if the orders have been passed on to the production and the date on which they were released.

	orderNo	productId	product	StartDate	Enddate	OrderType	Quantity	warehouse	released	releasedDate
14	PU-3017	7400-050	gearwheel, 50 tooths	21.07.2002 00:00:00.0000	29.07.2002 00:00:00.0000	PurchOrder	20	warehouse_of_material		
15	PU-3018	7400-050	gearwheel, 50 tooths	21.07.2002 00:00:00.0000	29.07.2002 00:00:00.0000	PurchOrder	20	warehouse_of_material		
16	PU-3019	7400-050	gearwheel, 50 tooths	21.07.2002 00:00:00.0000	29.07.2002 00:00:00.0000	PurchOrder	20	warehouse_of_material		
17	PU-3039	7300-060	round bar, 60 mm	21.07.2002 00:00:00.0000	29.07.2002 00:00:00.0000	PurchOrder	10000	warehouse_of_material		
18	PO-2021	2100-100	shaft 100 mm	28.07.2002 01:30:00.0000	28.07.2002 18:20:00.0000	ProdOrder	10	warehouse_of_products		
19	PO-2023	2100-100	shaft 100 mm	28.07.2002 03:20:00.0000	28.07.2002 20:10:00.0000	ProdOrder	10	warehouse_of_products		
20	PO-2026	2100-100	shaft 100 mm	28.07.2002 14:20:00.0000	29.07.2002 07:10:00.0000	ProdOrder	10	warehouse_of_products		
21	PO-2051	2100-200	shaft 200 mm	28.07.2002 17:15:00.0000	29.07.2002 14:15:00.0000	ProdOrder	10	warehouse_of_products		
22	PO-2020	2100-100	shaft 100 mm	28.07.2002 18:20:00.0000	29.07.2002 11:10:00.0000	ProdOrder	10	warehouse_of_products		
23	PO-2022	2100-100	shaft 100 mm	28.07.2002 20:10:00.0000	29.07.2002 13:00:00.0000	ProdOrder	10	warehouse_of_products		
24	PO-2052	2100-200	shaft 200 mm	28.07.2002 21:00:00.0000	29.07.2002 18:00:00.0000	ProdOrder	10	warehouse_of_products		
25	PO-2029	2100-100	shaft 100 mm	29.07.2002 01:20:00.0000	29.07.2002 18:10:00.0000	ProdOrder	10	warehouse_of_products		
26	PO-2053	2100-200	shaft 200 mm	29.07.2002 04:15:00.0000	30.07.2002 01:15:00.0000	ProdOrder	10	warehouse_of_products		
27	PO-2024	2100-100	shaft 100 mm	29.07.2002 05:20:00.0000	29.07.2002 22:10:00.0000	ProdOrder	10	warehouse_of_products		

A section of the order list

The checkbox "Plan after reset"

If the checkbox is set, then with each reset, the planning will be deleted, if you uncheck it, it will only deleted if something substantial changed in the model, like the master data or the forecast.

The ProductionManager 

The object *ProductionManager* manages the correct processing sequence. It dispatches the production orders and the processing stations. Depending on the dispatch rule you select, it sends the part to be processed on to the next station. For this reason the *ProductionManager* has to know all stations, including the operations it can execute, and all part types (*EntityNo*) to be processed.

Dialog

Double-click the icon of the *ProductionManager* to open its dialog. Enter the required data into the dialog, compare Dialog of the ProductionManager on page 8. Below we describe the features in detail:

Production Orders

Click the button Orders to open the table *ProdOrderL* of the *ProductionManager*. The object *MRP* automatically enters the production orders that the material requirements plan creates into this table. It contains these entries:

- Number of the production order, starts with "PO".
- Part number (*EntityType*) and part type (*EntityName*)
- Operations plan
- Bill of materials
- Priority
- *releaseDate*, including start time, of the production order
- *dueDate*, including the time at which the order is available in the warehouse
- *quantity* lot size of the production order
- *unit_of_quantity* of the production order
- *MasterData* warehouse that stores the material
- *start* time of the assembly process
- *Process* time of the processing process
- *end* time of the processing process

During your simulation run the *ProductionManager* enters the times into the last three columns. The column *start* contains the release date of the production order, for assembled parts, and the column *end* the point in time the product is completely assembled.

OperationsList

Click the button Show to open the list of operations, which the object *Validator* fills out. For each operation the program creates a table that contains all stations that execute this operation and a table of all parts waiting for this operation.

Stations List

This list contains the same information as the *OperationsList*, it is sorted by stations though. The program propagates information about operations of stations back to the stations. Whenever you change the contents of this list, you have to start the *Validator*!

Dispatch Rules

Click the button Select to open a dialog, where you select dispatch rules. You can select rules for dispatching the production orders, for selecting the stations and for selecting waiting parts. Compare the chapter *Production Order Control* for details.

Opening the Object

Click the button Open to open the *Frame* of the object. Here you can customize lists and methods of the *ProductionManager*. You might, for example, create dispatch rules for your modeling needs.

The Station 

Use the object *Station* to model machines and work stations, that process a part. Once the part has been processed in all steps of the work schedule, it is transferred on to the respective warehouse, where it is destroyed. The *Station* sends all data about capacities, availabilities and processing costs to the *ProductionManager*.

Dialog

Double-click the object *Station* to open its dialog, compare The simulation model depicting the production on page 5. You can select and/or enter these settings:

Station identifier

Enter or change the name of the object.

List of operations

Click the button Open to open a list of all operations of the *Station*.

Pace factor

Enter the pace factor. It allows you to differentiate between older and newer machines. Time calculation also uses this factor.

Tab Costs

Select the check box so that you can enter the relevant data.

Cost factor per hour

Enter the cost factor per hour for each processing step into this text box. The *Station* saves it in the variable *costFactor*.

Cost center

Enter the name of the cost center. This way you can assign several stations to a single cost center. When you use the object *Analyzer*, you can gather the processing costs for each station.

Tab Failures

Model machine failures, such as failures or maintenance, with failures of the *SingleProc Work* within the Station. The *ProductionManager* does not send any parts to a failed station. Enter the failure times into the dialog of the *Station*.

Select the check box in the dialog of the *Station*, if you want to simulate failures or not. When you select the check box, the program activates the text boxes MTTR and MTBF. Once you enter two of the values, the program calculates the third value.

You can assign a state icon to a failed station that designates the failed state. The simulation shows this icon during a failure, when:

- The name of the icon ends in *_fail*, and
- The remainder of the name is the same as that of the not failed *Station*.

The StationsProtocol

The stations protocol records the data of the production orders that the Station processes. It records and shows:

- The name of the production order.
- The name of the part.
- The lot size.
- Start time and duration of set-up.

- Start time and duration of set-up.
- End of processing.

The StationPool 

The *StationPool* has the same functionality as the *Station*. It consists of several *Stations* arranged in parallel. The *StationPool* records the utilization of each *SingleProc*, the *StationProtocol* contains the data of all stations.

Use the *StationPool* to simulate several *Stations* without having to enter the same values multiple times.

Dialog

The *StationPool* has the same dialog as the *Station*, we just added a text box into which you enter the number of stations.

The WarehouseManager 

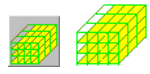
Use the objects *WarehouseManager* and *Warehouse* to model a warehouse system. The *WarehouseManager*:

- Manages moving parts into the warehouse and out of it.
- Checks availabilities before starting a production order.
- Records missing production orders and releases waiting orders.

Dialog

Click the button Show to open a list of all warehouses inserted into your model. Double-click the right part of a cell containing an entry to open the dialog of that warehouse. Before starting a production order the program checks, if the amount of items contained in the bill of materials is available in the warehouses.

Click the button Show to show all currently missing purchase parts. Click the button Open to open the *Frame* of the *WarehouseManager*, compare Dialog of the WarehouseManager on page 10.

The Object Warehouse 

The object *Warehouse* stores the parts that are processed completely. Double-click its icon to open the dialog of the object *Warehouse*.

- It manages the available stock, which is contained in the column stock, and the reserved stock for the part types (*EntityNo*) in the table *storeL*.
- Informs the *WarehouseManager* about missing parts (table *stockOutageL* of the *WarehouseManager*), when a part type for a production order is missing.

Dialog

Initial Stock

Click the button Show to open the table *WarehouseManager* and enter the initial stock at the start of the simulation. The contents of this table do not change during the simulation.

Current Inventory

Click this button to open a table that shows the current inventory after you stopped the simulation or after it has been finished. The stock is split up in available stock and reserved stock.

Tab Costs

Calculate costs

Select this check box to activate the calculation of costs.

Cost factor per hour

Enter the cost factor for transfer to stock and release from stock. The respective warehouse stores this value in the Variable *costFactor*.

Cost center

Enter the name of the cost factor. This way you can assign several warehouses to a single cost factor. You can then use the object *Analyzer* to analyze costs.

Tab Protocol

Logging the Activities of the Warehouse

Enter the *productid* of the part you want to watch. Click the button Show to open a *Chart* showing the occupancy of the warehouse.

Tab Definitions

Defining the Entrance Buffer

Click the button Entrance Buffer to open the dialog of the built-in entrance buffer of the *Station*. Here you can dimension the entrance buffer.

Defining the Exit Buffer

Click the button Exit Buffer to open the dialog of the built-in exit buffer of the *Station*. Note that we used a *ParallelProc* to model this buffer. For this reason parts, and orders, can pass each other.

The PurchaseManager 

The object *PurchasingManager* manages all purchase orders and starts them so that they are ready on their delivery date, just as the *ProductionManager* does for the production orders.

Dialog

Double-click the object *PurchaseManager* to open its dialog, compare Dialog of the Purchasing Manager on page 11. The dialog of the object *PurchaseManager* provides these dialog items:

List of Purchase Orders

Click this button to open a table with this data:

- *OrderNo* order number, starting with PuO.
- *EntityType* and *EntityName*, entity number and entity type.
- *releaseDate*.
- *dueDate*, delivery date of the material by the supplier.
- *quantity* lot size of the production order
- *Uof_quantity* unit of quantity.
- *warehouseType*, name of the warehouse, where the parts are stored.
- *ArrivalTime*.

- *Costs* for purchasing the parts.

Delayed Deliveries

Enter the portion of the *belated* orders on the *not belated* deliveries into this table.

Maximum delay of delivery

Enter the maximum deferment of delivery in hours. The program calculates the deferment with a uniform distribution between 0 and the maximum delay of delivery.

The CustomerOrder 

The object *CustomerOrder* simulates the income of orders from customer, that is the withdrawal of products from the warehouse. It is possible to generate the orders from the forecast or to enter them directly.

List of customer orders

Click this button to open a table with this data:

- *OrderNo* order number, starting with "*CO*".
- *EntityType* and *EntityName*, entity number and entity type.
- *releaseDate*, the date where the order comes from the customer
- *scheduledArrifalTime*, delivery date of the material to the customer
- *quantity* size of the customer order
- *released*, indicates if the order is already in the system.
- *ArrivalTime*.
- *Delay*

The order number

The number will be generated by the system in its init phase. It is the system id of the customer order.

The entity type and name

In these two columns the identifier from the master data list is needed, at least the productId must be given.

The release date

At this date the order is known to the system (for correlating planning and actual condition).

The scheduled arrival time

Then the customer order is executed and wants to withdraw the products from the warehouse.

The quantity

Number of parts which is ordered.

Note: The following columns are filled from the system during a simulation run.

Released

Indicates if the order is already in the system

The arrival time and the delay

This is the date where the parts are withdrawn from the warehouse and extracted from the system, also the delay to the scheduled arrival time.

Use Forecast

If you want to transfer the forecast to the customer orders you can use this functionality. It is also possible to alter the data of the forecast.

Variation of Forecast

This is the function to alter the forecast. The method gets the productId and the quantity given in the forecast, and the result is the new quantity.

The TransportManager 

Use the objects *TransportManager* and *DockingStation* to model a transport system. Double-click the icon of the object to open its dialog window, compare Dialog of the TransportManager on page 11. Enter the number of *Transporters* used into the text box.

Click the button Edit to open a table containing the transport times, compare The transport table on page 12. The object *Validator* automatically enters data into this table.

The DockingStation 

Connect the *DockingStation* with the *Station*. Both work in conjunction with the *TransportManager*. The *Station* transfers parts from the exit buffer *Station* to the *DockingStation*. This creates a transport order in the *TransportManager*. When a *Transporter* is available, the *TransportManager* fulfills this transport order using the transport times When the *Transporter* arrives at the target station, the part is transferred into the entry buffer of the *Station* and the *Transporter* can fulfill the next order.

Note: To connect *Station* and *DockingStation*, drag the *Station* over the *DockingStation* and drop it there. As long the *Station* is not connected to a *Buffer* or some such object, the program connects it directly with the *DockingStation*. When the *Station* is already connected to another object, you have to manually connect it.

The Part 

The object *Part* models the part to be processed. It provides these custom attributes:

EntityType

This custom attribute, of data type *string*, is the identifier of the part type, such as "2100-100".

EntityName

This custom attribute, of data type *string* identifies the part type, such as "shaft 100mm".

OperationsPlan

This custom attribute, of data type *table*, is a copy of the work schedule.

OperationsPlanPos

This custom attribute designates the current processing state in relation to the work schedule. When the part is created, it has the value 1. After the processing step has been executed, the value will be increased by 1. It has the data type *integer*.

billOfMaterial

This custom attribute, of data type *table*, is a copy of the *EntityType*.

globDestination

This custom attribute saves the path to the next processing station. When the part arrives at the exit buffer of the Station, the ProductionManager determines the new target. It is of data type *object*.

OrderNo

This custom attribute, of data type *string* is the order number of the production or purchasing order.

OrderLPos

This custom attribute, of data type *integer*, saves the row number of the production order.

quantity

Each part stands for a lot. This custom attribute, of data type *integer* saves the lot size.

WarehouseType

This custom attribute, of data type *string*, saves the type of warehouse, that stores the *EntityType*.

startSetup

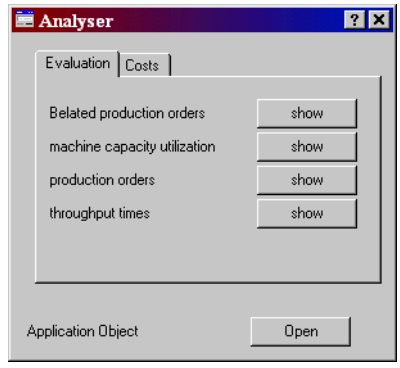
This custom attribute, of data type *dateTime*, saves the point in time where set-up starts.

The Analyzer  

During the simulation run, the *Analyzer* gathers all data required for evaluating it, and processes it.

Dialog

Double-click its icon, to open the dialog of the *Analyzer*. It provides its features on two tabs:



Tab Evaluations of the object *Analyzer*

Tab Evaluations

Delayed production orders

The *Analyzer* checks the delivery date for each production order. When the difference is greater than 4 hours, it inserts this order into the table *BelatedOrderL*, that lists the delayed production orders. Click the button Show to open a table and a chart showing the delayed orders.

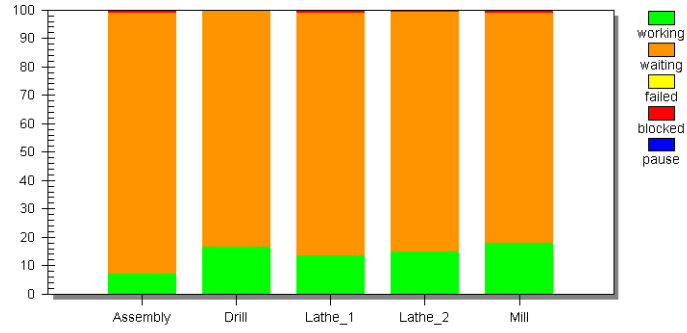
Standard Evaluation

The object gathers the data during the simulation. It analyzes these items:

- Utilization of machines

- Production orders
- Throughput times

Click the button Show next to Machine capacity utilization to open a chart showing the name of the station, the processing time, the waiting time, the pausing time, and the failure time, compare the screenshot below.



Utilization of machines

Click the button Show next to Production orders to open a table showing the values of interest.

	string 0	string 1	integer 2	time 3	time 4	time 5	time 6	time 7	time 8
string	OrderNo	EntityType	Quantity	Process	Set-up	Transport	Wait	Failure	Pause
16	PO-2401	PP-10	20	3:18:20.0000	0.0000	0.0000	1:12:17:42.500	0.0000	15:00:00.0000
17	PO-2402	PP-10	20	3:18:20.0000	0.0000	0.0000	0.0000	0.0000	1:00:00.0000
18	PO-2403	PP-10	20	3:18:20.0000	0.0000	0.0000	3:02:41:40.000	0.0000	0.0000
19	PO-2404	PP-10	20	3:18:20.0000	0.0000	0.0000	5:49:22.5000	0.0000	16:00:00.0000
20	PO-2405	PP-10	20	3:18:20.0000	0.0000	0.0000	21:47:17.5000	0.0000	2:00:00.0000

Table showing different times

This table shows the throughput times of all production orders with start time and end time.

	string 1	datetime 2	datetime 3	time 4
string	Auftragsnummer	Beginn	Ende	Durchlaufzeit
1	PO-2001	29.06.2001 14:26:40.0000	03.07.2001 09:02:30.0000	3:18:35:50.0000
2	PO-2002	28.06.2001 11:53:20.0000	02.07.2001 14:31:15.0000	4:02:37:55.0000
3	PO-2003	27.06.2001 10:20:00.0000	28.06.2001 07:47:05.0000	21:27:05.0000
4	PO-2004	26.06.2001 08:46:40.0000	26.06.2001 15:06:40.0000	6:20:00.0000
5	PO-2005	25.06.2001 07:13:20.0000	25.06.2001 15:33:20.0000	8:20:00.0000
6	PO-2006	30.07.2001 14:26:40.0000	31.07.2001 13:28:45.0000	23:02:05.0000
7	PO-2007	27.07.2001 11:53:20.0000	30.07.2001 10:52:05.0000	2:22:58:45.0000
8	PO-2008	30.08.2001 14:26:40.0000	03.09.2001 09:58:45.0000	3:19:32:05.0000
9	PO-2009	29.08.2001 11:53:20.0000	31.08.2001 14:10:00.0000	2:02:16:40.0000

Throughput times

The difference between both times is the throughput time. It is contained in the last column.

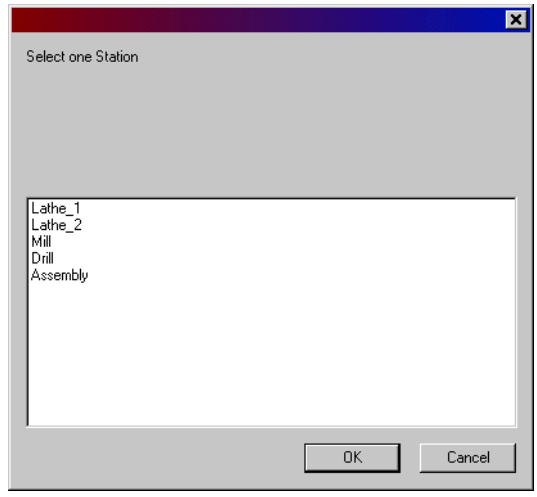
Tab Costs

The *Analyzer* provides four types of cost evaluations:

- Station costs
- Cost centers
- Order costs
- Mean costs of the parts

Station costs

When you select cost evaluation, the program opens the dialog below. Select a *Station* here. The program then opens a table showing all orders, compare the table *ProdOrderL*, with the respective processing and set-up costs, compare Section of the costing table of the production orders on page 26.



Dialog Select Station

Cost centers

After you selected the type of evaluation to be performed, select a cost center. This opens a table containing production orders, similar to the one shown below.

	string 1	real 2	real 3
string	Produktionsauftrag	Bearbeitungskosten	Rüstkosten
1	PO-2020	400.00	0.00
2	PO-2021	400.00	533.33
3	PO-2022	400.00	0.00
4	PO-2023	400.00	0.00
5	PO-2024	400.00	0.00

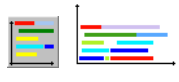
Section of the costing table of the production orders

Order costs

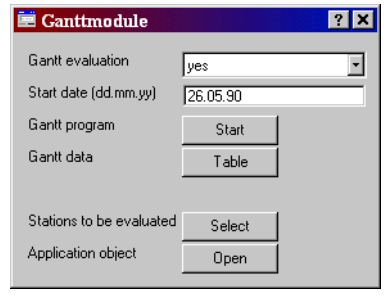
	string 1	real 2	real 3
string	Station	Bearbeitungskosten	Rüstkosten
1	Lathe_1	300.00	0.00
2	Lathe_2	800.00	0.00
3	Mill	400.00	0.00
4	Drill	275.00	0.00

Sample of a cost center of a production order

After you selected a production order, the program opens a table containing the name of the station, the processing costs and the set-up costs.

The Object Gantt 

Double-click the object *Gantt* to open its dialog.



Dialog of the object *Gantt*

The object *Gantt* contains all methods and tables required for creating a Gantt chart visualizing the simulation results. Use the Gantt chart to visualize the production data with start time and the sequence of processing steps on the machines.

Gantt Evaluation

Select Yes on the drop-down list Gantt evaluation to start collecting Gantt data. When an object executes an action, it enters times into the data table of the object *Gantt*.

Start date

Enter the time the object starts collecting data. Note that you have to enter the time before starting the simulation.

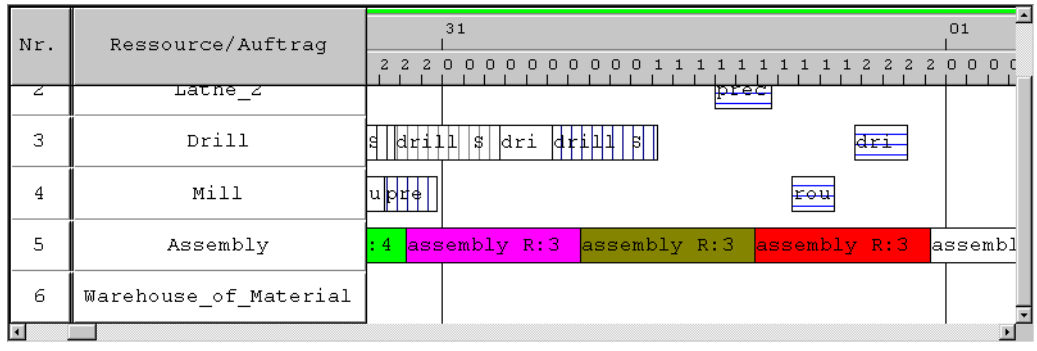
Gantt program

Click the button Start to show the Gantt chart.

Gantt data

Click the button Table to open the Gantt data table, compare the screenshot below.

	integer 0	string 1	string 2	datetime 3	datetime 4	string 5	string 6	integer 7	real 8
string	Resource No	Ressource	Auftrag	Startdatum	Enddatum	Operation	Teile-Nr.	Ebene	Losgroesse
1	7	Warehouse	PO-2021	28.07.2002 01:30:00.0	28.07.2002 01:30:00	warehouse	2100-100:1	22	10.00
2	7	Warehouse	PO-2023	28.07.2002 03:20:00.0	28.07.2002 03:20:00	warehouse	2100-100:2	24	10.00
3	1	Lathe_1	PO-2021	28.07.2002 03:30:00.0	28.07.2002 07:30:00	rough turn R:28	2100-100:1	22	10.00
4	1	Lathe_1	PO-2023	28.07.2002 07:30:00.0	28.07.2002 09:30:00	rough turn R:28	2100-100:2	24	10.00
5	2	Lathe_2	PO-2021	28.07.2002 08:30:00.0	28.07.2002 13:10:00	precision turn S	2100-100:1	22	10.00
6	7	Warehouse	PO-2026	28.07.2002 14:20:00.0	28.07.2002 14:20:00	warehouse	2100-100:3	27	10.00



Gantt data table and how it is displayed

The columns show these values:

- Column 1: Station (Resource)
- Column 2: Production order

- Column 3: Start of processing
- Column 4: End of processing
- Column 5: Operation being processed
- Column 6: Part number, compare *masterdataL*.
- Column 7: Color of the bars, used for distinguishing the production orders
- Column 8: Lot size

Editing the Station List

Click the button Select to open a dialog and select the stations to be watched there.

Stock Evaluation 

The stock evaluation shows the stock located in the warehouse and the circulating stock. You can view the one type of data only or accumulated data, you can view it from a certain point in time on, or for a certain duration.

Dialog

Dialog of the *StockEvaluation*

Consider inventory of warehouse

Select the check box to consider the stock in the store in addition to the circulating stock. Do not select it to just show the circulating stock.

Accumulated Material

Select the check box to show the accumulated number of parts. Do not select it to show the parts split up by part types.

Select Material

Click this button and select the types of parts that are to be considered.

Select the Interval to be Viewed

On the drop-down lists you can select the starting time and the ending time of the interval you want to view.

Start collecting stock

Enter the point in time from which on the *StockEvaluation* starts recording data.

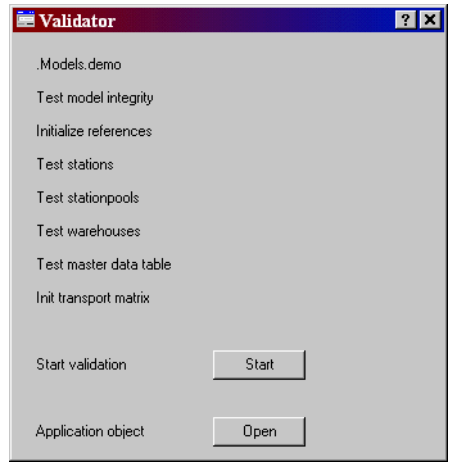
Interval

Enter the interval how much time has to pass between recording stock.

The Validator

The object *Validator* checks the consistency of your simulation model and enters data into several tables. You can start it:

- Directly in your model, after you made changes to it and then reset it.
- In the dialog of the object *MaterialRequirementsPlanning* by selecting *Test the model*.
- By selecting it in the class library and then selecting your model in the dialog that opens.



The dialog of the *Validator*

Resetting your model after changing it automatically starts the *Validator*. In the other cases you have to manually start the test functions.

When the operations have been executed correctly, the dialog shows an OK after the operation.

Checking the integrity of the model

The *Validator* checks if all objects required for the functionalities of the model are inserted into it. If this is not the case, it inserts them for you. It also checks if objects that are needed once only were inserted twice.

Initializing references

The *Validator* initializes the references of the objects in the model to each other for accelerating the method calls.

Testing stations

The *Validator* enters all operations of the stations into the *ProductionManager*. It also checks in the *ShiftCalendar*, it checks if operations exist and if the speed factor you selected makes sense.

Testing station pools

Performs the same tests and operation as for the stations, just for the *Stationpools*.

Checking warehouses

The *Validator* registers the warehouse with the *WarehouseManager* and the *ShiftCalendar*.

Testing the master data table

The *Validator* checks if the data in the master data table is correct, i.e. if all products in the bill of materials are contained, if all operations are available, and if all required fields are filled out.

Initializing the transport tables

The *Validator* enters the indexes of the transport table when you inserted a *TransportManager* into your model.

4

The Internal Control of the Production

Chapter 2 explained how to fill out order lists, operation lists and master data lists. It also described how to set set-up and processing times. Chapter 3 showed how to use the objects *MRP*, *ProductionManager*, *PurchaseManager* and *WarehouseManager*, *Warehouse* and *Station*.

Generating Production and Purchase Orders

When you modify the tables *masterData, forecast* or *productmix*, the production orders will also change. After a Reset the *Init* method calls the method *BuildProductForecast*, which fills the table *productionSchedule* with the planned orders for the products of the forecast. Then the method *HandleOrders* is called, that breaks down the bill of materials, that schedules the orders and creates the production and purchase orders according to the rules for the lot sizes. A loop covers the instructions in the table *productionSchedule*, then the method opens the product-relevant subtable and covers the orders contained in it. For a planned order this order is charged against the stock in the warehouse, and the amount of parts to be produced is determined: The amount using the lot size rules and the starting point in time of the orders using scheduling. For a production order derived planning orders are created for the components using the break-down of the bill of materials in the methods *getPartsList* and *InsertRequirements*. The method *TransferSchedule* then copies the production and purchase order from this product-oriented list into the time-oriented list *productionOrders*.

Dispatching Production Orders and Purchase Orders

During initialization the methods *startOrder* of the *ProductionManager* and of the *PurchaseManager* are called. A waituntil-statement instructs them to wait until they are assigned orders from the object MRP. During initialization of the *MRP* the method *release* is called. It calls itself in the interval you enter for the *Releaseinterval*. This method calls the method *releaseOrders*. It transfers all orders of the table *productionOrders*, whose *StartDate* is before the end of the next call, to the proper manager, it transfers purchase order to the *PurchaseManager* and production orders to the *ProductionManager*. When orders were transferred, the global Variable *working* notifies the respective Manager.

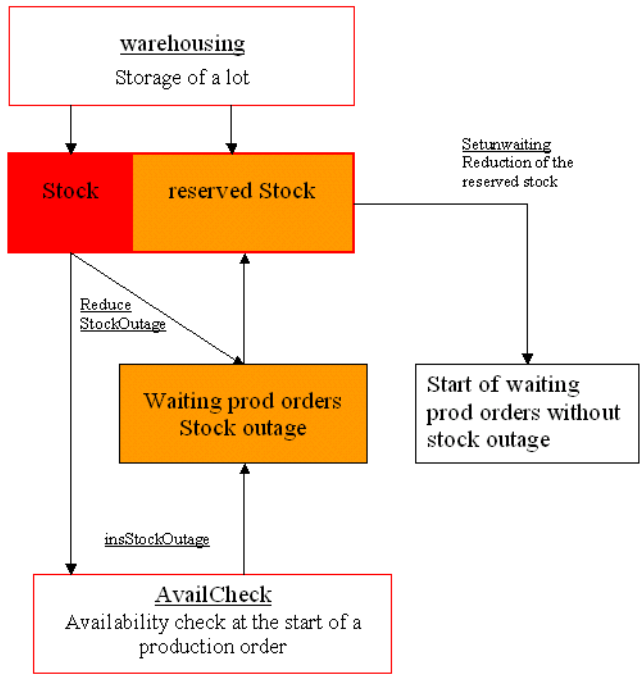
The *ProductionManager* embraces all orders with the same release date. It passes the order on to the rule you selected for scheduling the order. Orders for a later point in time are dispatched with the statement *ref(startOrder).methCall(release-SimTime)*. When only a single order is to be dispatched at the current simulation time, the method *enterOrder* is called immediately. The method *enterOrder* creates an MU and sets all attributes. Then the program determines the next target, which will be a warehouse.

Releasing Production Orders

When the *WarehouseManager* takes control over a production order, it checks the availability of all components of the respective bill of materials, using the method *availCheck* in the *WarehouseManager* itself and in the warehouses. This decreases stock in the warehouse and increases the reserved stock.

If not enough stock is available, the production order is transferred to the warehouse *waitingOrder*. The WarehouseManager enters missing parts into the table *stockOutage*.

As soon as enough stock for all components is available in the warehouse, the production order will be started. The method *WH_startPart* decreases the reserved stock.



Availability and reserved stock

The Kanban Control

You can also control the production according to the Kanban principle. Select Kanban as lot size rule for a product. This implies that all components of this product are controlled by the Kanban principle.

For a product created according to the Kanban principle no production orders are created during the planning phase. They are only created when the model runs. When the method *warehousing* of the responsible *Warehouse* detects that the amount of product falls short of the minimum amount, it automatically creates production orders.

This requires that you entered a fixed lot size, which corresponds to a production order/bin, into the master data table, as well as the amount of possible Kanban cards, which corresponds to the maximum number of production orders/bins. You can also enter the start time for a Kanban order.

Moving Parts On

When the MU leaves the processing station in the *Station* or in the *StationPool*, it calls the *ExitControl*. This method first increments the counter for the work schedule. Then, it registers the set-up time and the processing time for this station, and registers it with the *Analyzer*, if it exists. The method *nextDestination* of the *ProductionManager* determines the next target of this lot.

When the last processing step of the work schedule has been finished, the MU is moved on the appropriate warehouse. The method *warehousing* registers the part with the warehouse. Then, the method *deletePart* of the ProductionManager deletes it. The method *startOrder* of the *ProductionManager* tests, if all production orders have been started. It then waits and makes sure that all started production orders have also been deleted. Then, it stops the simulation.

The exit control calls the method *NextDestination* of the *Station*. It checks if the *Station* has a successor. When no successor exists, i.e. no docking station of a transport system is connected, the method *NextDestination* of the target station moves

the MU to the next *Station*. Otherwise, the MU is moved into the docking station of the transport system. This then moves the part on to the target station, which you entered into the attribute *globDestination*.

Transferring Orders to Stock

When a production or purchase order enters the entrance buffer *InWork*, the method *warehousing* is called. It searches for the *EntityType* in the missing parts list *stockOutage* of the *WarehouseManager*. When this list contains parts of this type, the method *ReduceStockOutage* decreases the amount of missing parts. The method returns the reduced amount of missing parts. The table *storeL* saves this amount as reserved stock. It saves the remainder as available stock. When no other entry for the stored part type exists, the method *delStockOutage* of the *WarehouseManager* starts the waiting orders. Otherwise the order has to wait for the next transfer to stock of the missing entity numbers.

Set-up and Processing Time

The program calculates the set-up and processing times using the formulas *calcSetupTime* and *calcProcessTime*. These formulas enter the order into the station protocol and calculate the respective times.

5

Production Order Control

Production order control and machine disposition enable you to easily determine the best control strategies for the production process. For this reason you do not have to implement your own control strategies.

We provide rules for entering orders at the start of the simulation (*Entry Queuing*), rules for transferring the parts to the next Station (*Process Sequencing*) and for selecting the stations (*Resource Selection*).

The rules for these three strategies are applied in an analogous way. You find the rules in the *Frames OD_Strategies, PD_Strategies* and *SD_Strategies*. They contain these components:

The table *StrategyL*. The column *Strategy*, that contains a short description of the rule. The column *Method* shows the path to the respective method. The column *Strategy* is used for the drop-down lists of the dialog Rules, compare Select dispatch rules on page 10.

The methods *orderDispo* (or *PartDispo* or *StationDispo*) are used by other methods. These methods only call the method you selected. The global Variable *actStrategy* points to the method you entered into the table *StrategyL*.

Proceed as follows to implement a new rule:

- Insert an instance of the object *Method* and implement the rule.
- Enter a brief description of the rule into column *Strategy* of the table *StrategyL*. Enter the name of the method into the column *Method*.

Now you have inserted your new rule! Double-click the icon of the *ProductionManager* to open its dialog. Click the button Select. Open the respective drop-down list. It will now show the name of the rule you just added.

Order Disposition

We implemented four rules for *Entry Queueing*. If more than one order has the same release date, one of these rules determine the sequence of the orders:

The first rule, *orderList* starts the order in the sequence they were entered into the production order list.

The second rule, *manPriorities*, uses the priorities entered into the production order list.

The third rule, *dueDate*, enters the order according to their due date.

The fourth rule, *random*, enters the orders in a random fashion.

All methods use the same arguments: *first_row* and *last_row* of data type integer, that point to the table *prodOrderL*. The method *enterOrder* enters all orders located within these two rows into the *ProductionManager*.

Part Disposition

Several parts waiting for an operation an available *Station* can execute are selected according to one of these rules:

The first rule, (*first suitable*, selects the first part from the list of waiting parts, using the FIFO rule.

There are two rules, *shortest waiting time* and *longest waiting time* which evaluate the waiting time of the part.

The rules *shortest operation time* and *longest operation time* select the part according to the required processing time, which is stored in the work schedule plan.

The rules *shortest order completion time* and *longest order completion time* select the part according to the time needed for the entire remaining processing time of the part which is determined by the operations plan and the processing state *operationsPlanPos* of the part.

The rule *delivery time* selects the part with the earliest delivery time contained in the production order list.

There also is a rule which selects the part randomly using a uniform distribution.

All methods use the same parameter list: The station which is available at the moment, and the list of waiting parts. The method returns the selected part (*object*).

Station Disposition

When a part is processed all the way by a station, the method *nextDestination* of the *ProductionManager* determines its next destination. This method uses the rules for determining the next station. We predefined 8 rules:

The first rule, *first suitable* selects the first available station.

The rules *minimum available capacity* and *maximum available capacity* select the station according to the available capacity in the entrance buffer.

The rules *minimum processing time* and *maximum processing time* select the station according to the processing time for the part which requests a destination. The processing time for the part is contained in the work schedule.

The rules *minimum tooling costs* and *maximum tooling costs* select the station according to the processing costs for the part which requests a destination. The processing time for the part is contained in the work schedule.

There also is a rule which selects the *Station* randomly using a uniform distribution.

The parameters for theses rules are the list of all possible stations and the part requesting a new destination. The methods return the selected *Station* (*object*). When there is no available station, the methods return the responsible *WarehouseManager*.

Synchronous Control

Another way of scheduling machines are fixed lines. Plant Simulation Shop supports this also. You will have to define it for the entire plant. Activating the synchronous control deactivates machine disposition. The table has to be unique for all combinations.

Enter the sending station into the left column of the table, the succeeding station into the right column.

6

Extensions

You can use Plant Simulation for modeling complex production systems. The universality of the library allows for modeling production systems in a wide variety of industries. You will also have to implement special features yourself. Below, we describe some extensions.

Connecting the Stations to a Transport System

For modeling transport systems, we provide the Plant Simulation library for material flow equipment.. The objects of this library enable you to realistically model the flow of materials between the processing stations in your factory. The Plant Simulation Shop *ProductionManager* determines the next target station of the part and saves it in the attribute *globDestination*. The transport system uses this to determine the available target, that is saved in the attribute *locDestination*. In general this docking station that connects the processing station and the transport system. Using the speed and the lengths of the tracks, you can realistically model the transport times between the stations.

Custom Lot Size Rules

You can also use your own lot size rules. For this to work, you have to enter the custom rule into the table *Rules* of the object *lotsizerules*. It is located in the folder *Internal* within the folder *Logistics*.

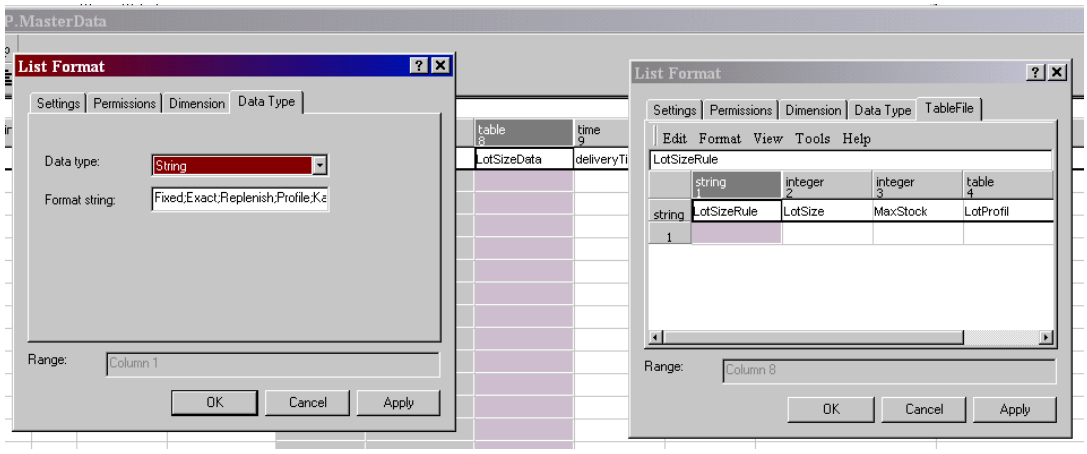
	string 0	object 1	string 2	stri 3
string	Rule	Method	Description	
1	Fixed	fixed	fixed lotsize is given	
2	Exact	exact	the production order has exact the same quantity as required	
3	Replenish	replenish	the production order quantity is the difference between max and current quantity in stock	
4	Profile	profile	the quantity of the production order depends on a custom defined profile	
5	Kanban	kanban	Kanban controlled.	
6				

The table *rules*

Enter the identifier of the rule, as it is going to appear in the master data table, into the row index. Enter the name of the method that is going to be called into the first column. Enter a short description of the rule into the second column.

The method has the arguments *productId* and the required amount of parts. It returns a list of integer values. The dimension of this list is the number of lots. The contents of the list is the size of the individual lots.

Finally you have to enter the rule into the master data table as one of the available rules. Or you have to cancel the rule constraints of the master data table. For this you either cancel or extend the constraint of the string values in the format table of the column lot size of the master data table.



Reformatting the master data table

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