Cost and Price Calculations

Produmex Manufacturing is a legacy product and Boyum IT Solutions no longer sells new installations for it.

Cost and price calculation is an integrated part of Produmex Manufacturing. When calculating the costs and prices of an own manufactured product the prices of the materials are simply added as much quantity is used for the production. Calculating the costs of operations and additional costs including energy, management, amortization, wages and so on is not as straightforward as calculating the material costs.

SAP Business One has basic machinery for manufacturing cost calculations: each item in the component list of a Bill of Material (BoM) or production order may have a price and SAP Business One calculates the total cost of a product by summing the products of item prices and quantities. If more sophisticated cost calculation is needed, instead of adding cost rows in BoMs, the built-in cost calculation facilities of Produmex Manufacturing should be used.

Manufacturing cost calculation starts by defining cost types that are used for any production operation in the company. The cost types are assigned to manufacturing resources (work center groups, work centers, operations, and so on) with their basic cost values. The costs of operations of in BoMs and Production Orders are calculated according to the ratio they use these resources. Manufacturing cost calculation is applicable only for manufactured products and components/parts (with procurement method “Make”). The actual algorithm of calculating the cost of a manufactured product is defined in cost schemas. The cost values calculated with cost schemas may be used to update the prices of the products in the pricelists. For this job price (calculation) schemas and intermediate cost collectors are needed. Costs can be calculated from BoMs and Production Orders. When the costs are calculated from Production Orders, the resource consumption values may come from the (1) planned component list, (2) the released component list where the operations are linked to specific work centers, (3) and the actual resource consumption reported via Production Data Collection (PDC).

Please note: The cost and price calculation logic of Produmex Manufacturing does not support the '*Remove Unpriced Items from Price List in Database*' setting. Make sure that the '*Remove Unpriced Items from Price List in Database*' setting is not enabled on the Pricing tab of General Settings in SAP Business One.

1. Defining Costs

Since calculation is sensitive to numerical precision, it’s highly recommended to increase the decimal places for Amounts in the General Settings form.

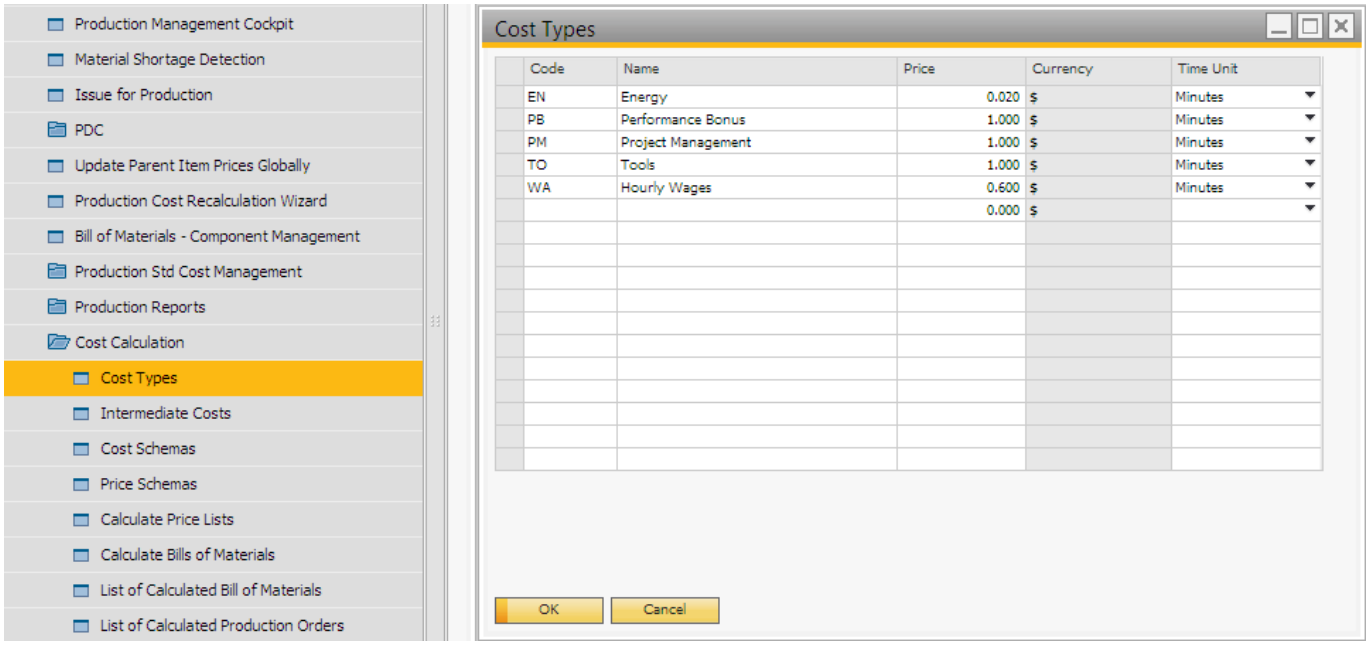
1.1. Defining Cost Types with Base Price

Cost types are the basis for all manufacturing cost and price calculations.

The code is an at most 8 character identifier.

Some costs are dependent on the length of a manufacturing operation (job) or the duration of the using of a resource (machine). For them the time unit can be defined for the base price. This price is a company-wide generic (base) price of the cost type for the selected time unit. The Energy, for example, in our sample company is $0.02 per minute; that is, 1 (kilo)wattminute of electricity costs $0.02 for our company.

The actual unit of measurement can be meant anything: megawatt-minute, kilowatt-minute, watt-minute, etc; what is important is that the price should be a ratio of the selected time unit. Later when the Energy costs are defined for the manufacturing operations or resources, the amount of energy that the operation/resource consumes in the selected time unit (minutes, in our example) should be defined.



Some other costs are not dependent on the duration of job/usage. For these cost types the time unit is not relevant. Most typically the price for them is set to $1 meaning that the actual cost prices will be defined later when the cost type is associated with a manufacturing operation or resource. Never define 0 as the price for a cost type, unless you want to have the system entirely ignore that cost type.

$1 can be defined for time dependent cost types as well, where no appropriate (relevant) company-wide price value is available or can be determined. The Hourly Wages in our example is an overall $0.6 per minute for every type of jobs in the company. If the price of wages a minute were different for each operation, then the price value of the cost type Hourly Wages should be defined as $1, and the actual minute-wages should be defined for each operation. The price value defined for a cost type is always multiplied with the cost amount defined for an operation or resource.

For the sake of understanding the cost types here are defined for minutes. In the example we define a

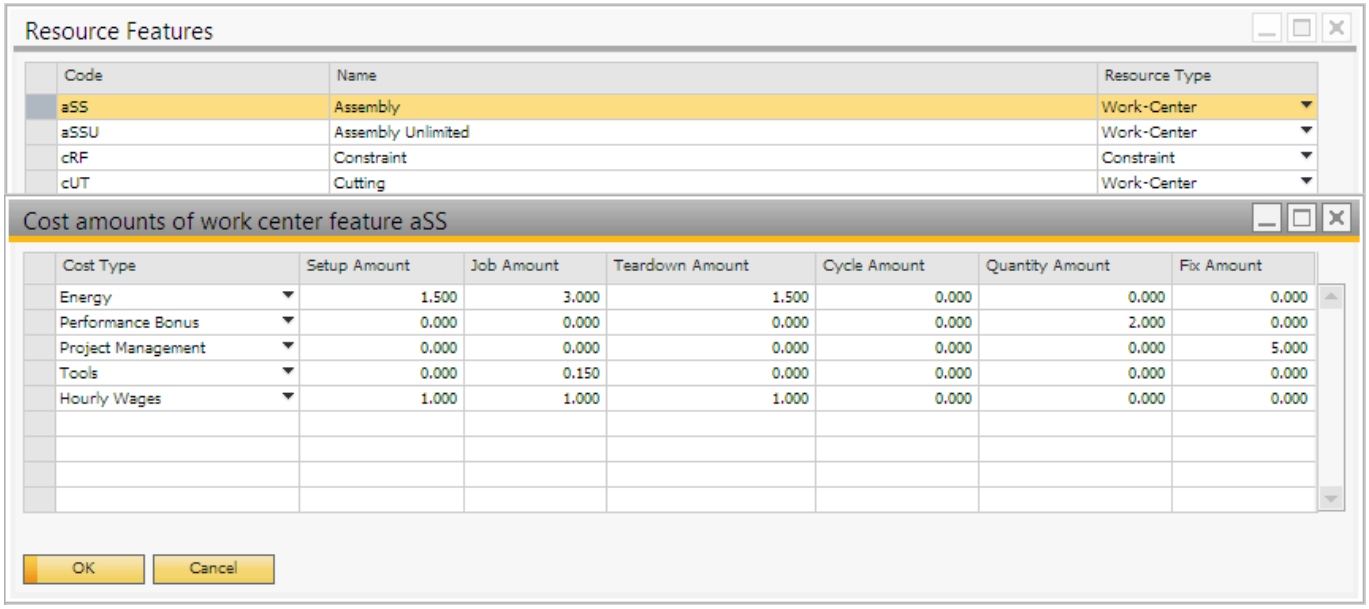
number of cost types:

- **Hourly Wages** – the cost of labor per minute.
- **Performance Bonus** – the bonus for a completed unit of work. For this cost type in this example the time unit is not relevant. We'll see later how it is used to define the actual bonuses when the types of jobs are defined. Whenever the price cannot be defined in general for the cost type, the value should be set to 1.
- **Energy** – the price is the minute cost of one unit of energy. This price is the current price of one unit of energy. Since our example company uses only electrical devices, the number is the price of one Watt-minute. Later we'll define the actual energy consumption of the tools.
- **Tools** – this is an estimated cost of the devices used for manufacturing. These costs will be defined later for the types of jobs.
- **Project Management** – each job should be communicated to the workers and instructions should be given. We'll define the management costs for the type of operations as fix costs. The Time Unit is not relevant for these (fix) type of costs.

All the cost types used for any manufacturing operations or resources of the entire company should be defined here. The cost types are not automatically associated to any of the manufacturing operations or resources; they must be explicitly linked to the resources and/or operations as described in details in the forthcoming sections.

1.2. Defining Resource Cost Amounts for Types of Jobs

A topmost level where manufacturing cost amounts can be defined is when Features are specified. In the Produmex Manufacturing add-on Features is a notion to define types of jobs, groups of machines, workers with the same skills, and so on.



In the example above the number of costs are defined for the Assembly feature:

- **Hourly Wages** – every minute (since we defined this cost type for minutes) of this (Assembly feature) job requires one minute of work from a worker with the minute price defined for the cost type. That is, for example, if 100 minutes (job time) of Assembly feature operation were required for a manufacturing the Hourly Wages cost would be 100 x 1 x 0.6 (from cost type

definition above). If 10 minutes of setup is defined for the operation 10 x 1 x 0.6 is calculated.
- **Performance Bonus** – for every completed unit of job 2 x 1 (from cost type definition) money is paid to the workers. Now it's getting clearer why the Price was set to 1 when this cost type was defined: the cost type amounts defined for the resources (features, work centers, operations) are multiplied by the base cost type price.
- **Project Management** - This cost is non-variable: 5 for each job taken. This is the cost of job administration. It is not dependent on the amount of job time. If a BoM had ten operations of this type, the total project management cost for the entire process would be 50 (10 x 5).
- **Energy** – every minutes, when this (Assembly feature) job is done on any work center (machine), 3 units (Watt-minutes) of energy is used. For Setup and shutdown the energy consumption is much less. The cost for 100 minute job would be 100 x 3 (job amount) x 0.02 (base price of cost type defined earlier).
- **Tools** – the cost of tools have been estimated to be 0.15 for each minute of this job. The total tools cost for a 100 minute job would be 100 x 0.15 (job amount) x 1 (base price of cost type defined earlier).

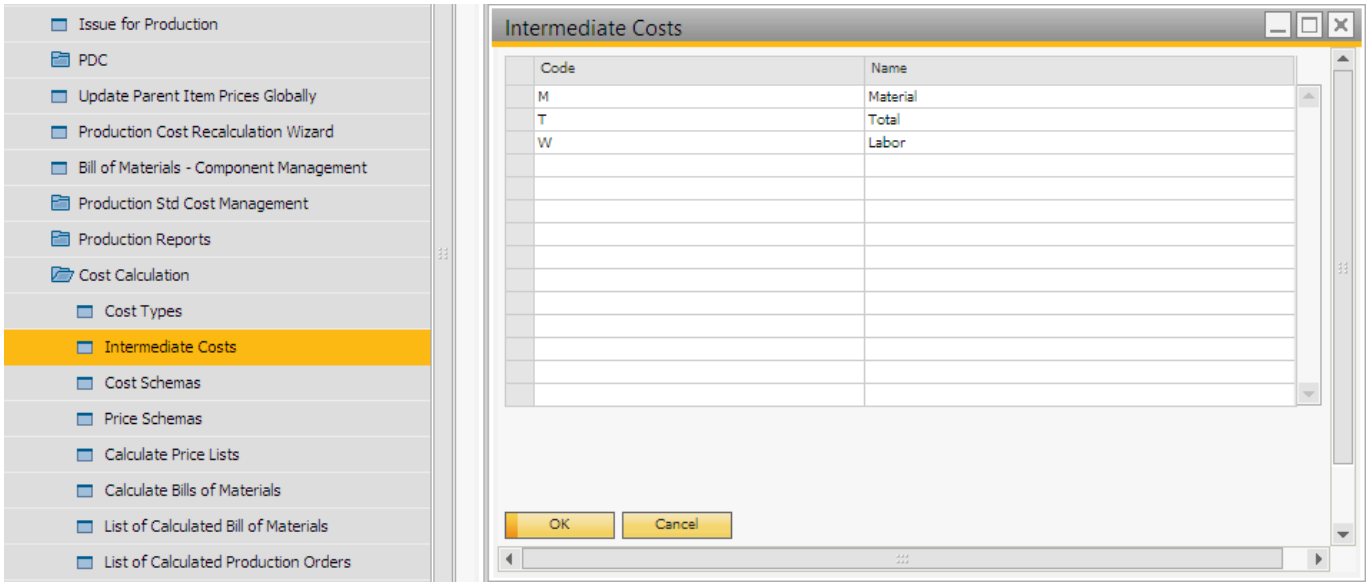
It's not necessary to define all cost types for all Features; for example, Tools and Energy may be insignificant for Quality Inspection.

It is possible to override these cost type amount values for specific work centers, operation master data, operations in bill of materials, operations in production orders.

1.3. Defining Intermediate Costs

An intermediate cost is actually a predefined name (variable) that can be used in calculation schemas. Intermediate costs are necessary for Price Schemas; intermediate costs are the linking machinery between Cost Schemas and Price Schemas. From the perspective of the calculation engine, Intermediate Costs are a kind of variables, when the engine executes/processes a cost schema it calculates the values and stores them in Intermediate Cost variables as defined in the Cost Schema.

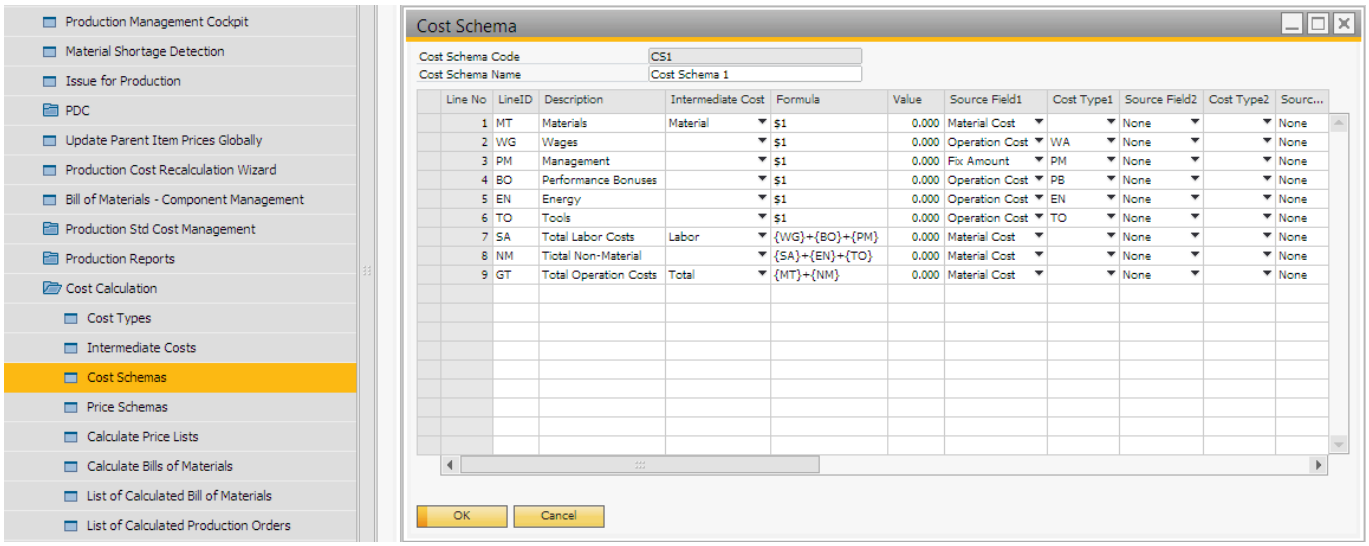
Normally, the calculation of Cost Schemas is followed by the calculation of a Price Schema. Price Schemas contain references to Intermediate Costs, and the values are coming from the calculated results of Cost Schemas.



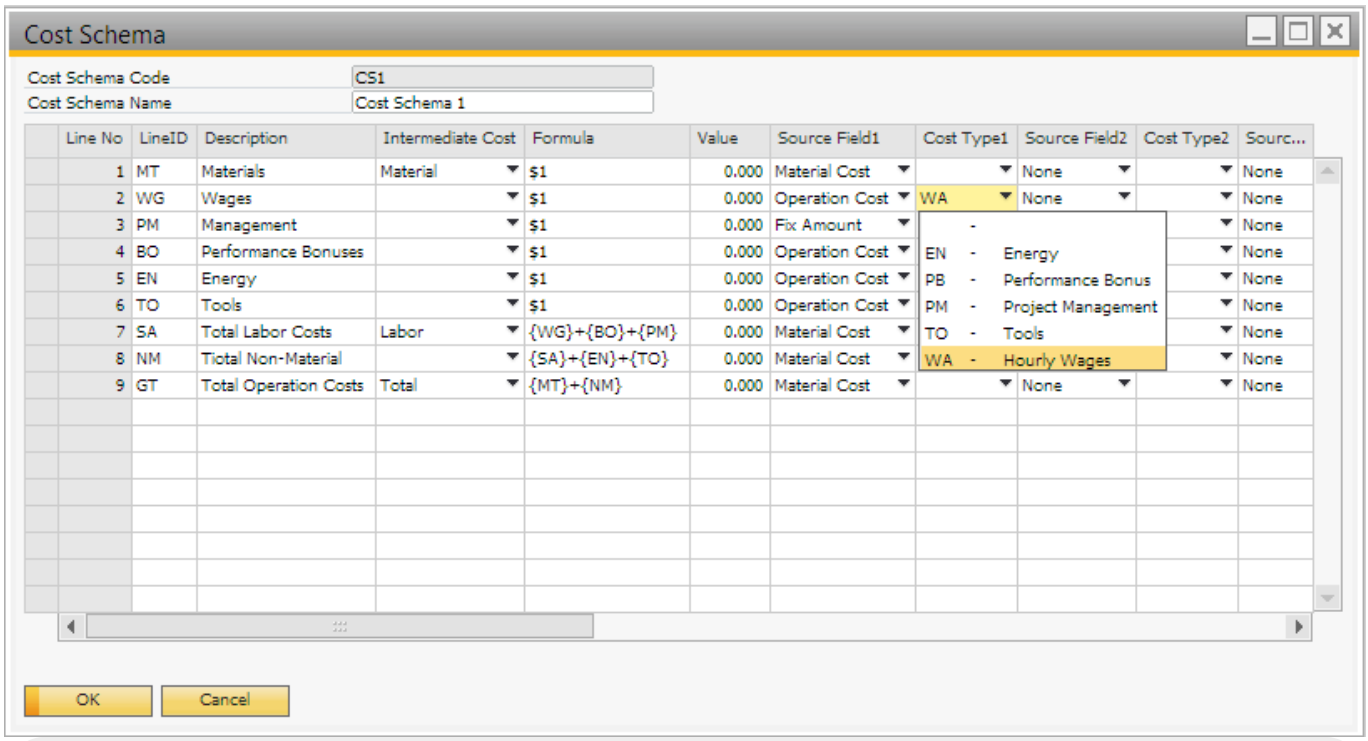
In our example we define only a couple of Intermediate Costs.

1.4. Defining Cost Schemas with Intermediate Costs

A Cost Schema is used to define the calculation of manufacturing costs for products with BoMs. A schema consists of lines; each line will have a value as defined by the Formula field when executed.



In our example we define a line for collecting the costs of purchased materials. The $1 in the Formula cell refers to the value in “Source Field 1”. Up to nine source fields (Source Field 2, Source Field 3, and so on) may be defined for a line and these values can be referenced with the symbols $1, $2, …, $9. The calculation engine has a number of predefined values that can be used as data source.



Material Cost is the price of a purchased material component. The calculated value of this cost schema line is saved in the intermediate cost variable “Material”. We will see later how this variable is used in a price schema.
In our sample we collect the values for each cost types.

The **Operation Cost** is the total amount of the operation lines in BoMs; for purchased material lines this value is 0. The Cost Type1 is a filter for Source Field1. Each source field has a corresponding filter field.

In our sample the **Wages** cost schema line is the sum of the Hourly Wages defined for operations. We have already explained how operations are related to cost types.

In the **Management** line we collect the costs of the cost type Project Management. Since this type of cost does not depend on the volume of the work, the Fix Amount value should be selected as source field.

In the **Total Labor Cost** line we sum the values from lines {WG} + {BO} + {PM}. The calculated value is saved in the intermediate cost (variable) “Labor”. In the Formula fields the previous lines can be referenced either with the line number or with the line ID.

The fields in the cost schema are the following:
** // Line No// **
The line number of the calculation row.

** // LineID. // **
The ID given by the user with which it is possible to refer to the line.

** // Description // **
The textual description of the calculation line.

** // Intermediate Cost // **
The type of the intermediate cost can be given here.

** // Formula // **
You can here set the formula according which the system should calculate. You can use these symbols:

- $: The value that is referred in the previous column.
- $x: Reference to source field (for example $1, $2, etc.). There are 10 source fields in the window; you can use the numbers 0-9 to refer to them.
- {LineID} = The result of the line with the given 'Line ID'.
- [LineNo] = The result of the line with the given 'Line No.'.
- [-x]: the result of the line which is x lines above this one.
- x%: A value in percent.

Please note: The only decimal separator supported is the dot (**.**).

Example:

Description	Formula	Source Field 1
Materials	$1	Material Cost

The material cost is the price of the purchased material components.

Use the DataTable.Compute method to create a more advanced formula. For more information about the method please see:

- Expression
- Compute

Example:

Description	Formula	Source Field 1	Source Field 2
Materials	IIF ($1>2, $2, 1.2*$2)	Quantity Produced	Material Cost

If the produced quantity is greater than 12, the material cost is the price of the purchased material components otherwise the material cost is the price of the purchased material components multiplied by 1.2.

MSSQL TIP
If the desired formula cannot be defined with the DataTable.Compute method, you can use SQL syntax to define the formula as well. Please keep in mind that defining several formulas with SQL syntax can lead to slower performance due the higher number of the SQL queries that the system executes.

Please note: Defining the formula with SQL syntax is not yet supported on HANA. Always use the DataTable.Compute method to define the formula.

Example:

Description	Formula	Source Field 1	Source Field 2
Materials	CASE WHEN $1> = 12 THEN $2 WHEN $1 > = 6 AND 12 > $1 THEN $2*1.25 ELSE $2*1.5 END	Quantity Produced	Material Cost

If the produced quantity is greater than or equal to 12, then the material cost is the price of the purchased material components. If the produced quantity is less than 12 but greater or equal to 6, then the material cost is the price of the purchased material components multiplied by 1.25. If the produced quantity is less than 6, then the material cost is the price of the purchased material components multiplied by 1.5.

** // Value // **
A set value which will be used in the formula given in column Formula.

** // Source Field 0-9 // **
The source fields which can be used by calculation. These can be:

- **Calculation Base Quantity**: the calculation base quantity given in the head of the BoM.
- **Setup Time**: the Setup Time from the BoM lines.
- **Job Time**: the Job Time from the BoM lines.
- **Teardown Time**: the Teardown Time from the BoM lines.
- **Purchasing Price**: the price based on the price list set in the BoM lines.
- **Inhouse Price**: the Inhouse Price is the price of the product calculated recursively based on the BoM. It means in case of produced goods the program calculates the costs of all raw materials that are in the BoM of the produced good, and the costs of the raw materials will be calculated according to their own calculation schema (for produced goods it will be calculated based on the BoM, for purchased goods the price will be taken from the set price list). The prices of all raw materials are then summed, and if there is an outsourced part of the production, then the in house part will be calculated, and this will be the inhouse price.
- **Outsourcing Price**: The price of the item based on the price list set in the BoM and calculated for the outsourced quantity.
- **By Product Price**: the price of the by-product based on the price list set in the BoM.
- **Quantity Produced**: The Quantity Produced from the BoM.
- **In House Quantity**: The In House Quantity from the BoM.
- **Outsourced Quantity**: The Outsourced Quantity from the BoM.
- **By Product Quantity**: The By Product Quantity from the BoM.
- **Purchased Quantity**: The Purchased Quantity for purchased good from the BoM.
- **Setup Amount**: the Setup Amount for the given resource from the cost amounts.
- **Job Amount**: the Job Amount for the given resource from the cost amounts.
- **Teardown amount**: the Teardown Amount for the given resource from the cost amounts.
- **Quantity Amount**: the Quantity Amount for the given resource from the cost amounts.
- **Fix Amount**: the Fix Amount for the given resource from the cost amounts.
- **Cost Type Price**: the price from the cost amounts (the contents of the Price column in cost amounts).

- **Cycle Amount**: the Cycle Amount for the given resource from the cost amounts.
- **Cycle Count**: the number of setup and teardown cycles (quantity produced/calculation base quantity).

In addition there are calculated fields, the calculation is the following:

- **Setup Cost**: Setup Time * Cycle Count * Setup Amount * Unit Price
- **Job Cost**: Job Time * Quantity Produced * Job Amount * Unit Price
- **Teardown Cost**: Teardown time * Cycle Count * Teardown Amount * Unit Price
- **Cycle Cost**: Cycle Count * Cycle Amount * Unit Price
- **Quantity Cost**: Quantity Produced * Quantity Amount * Unit Price
- **Operation Cost**: Setup Cost + Job Cost + Teardown Cost. + Cycle Cost + Quantity Cost.
- **Fix Cost**: Fix Amount * Unit Price //Where the Unit Price is the price from cost amounts. //

These costs are calculated for both the head and all lines.

Other calculated fields:

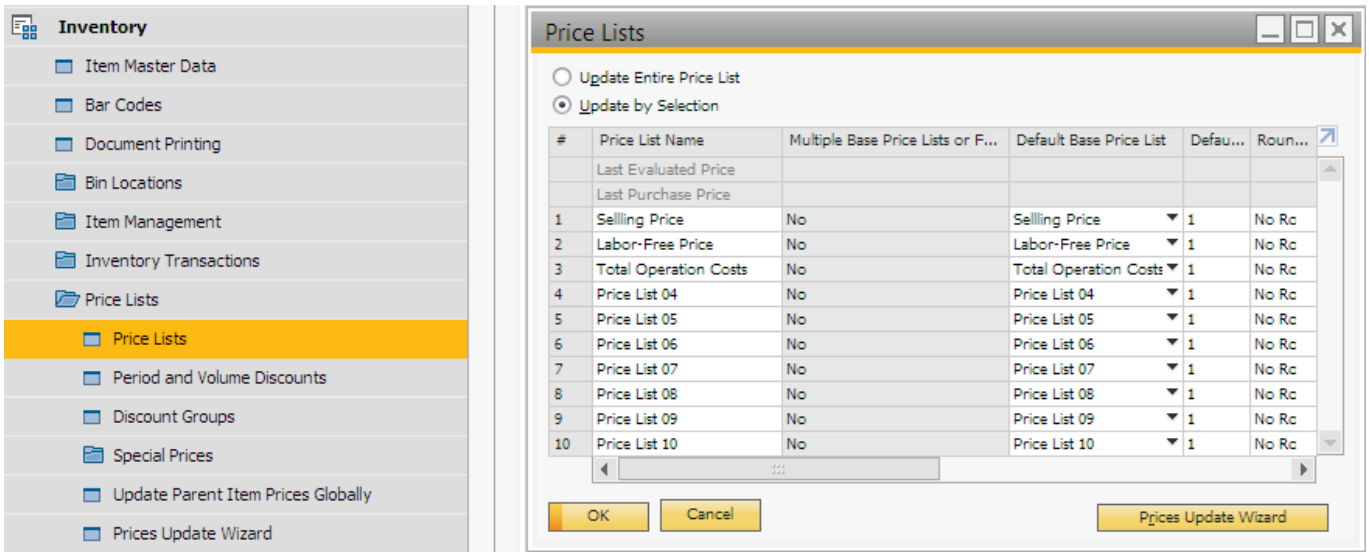
- **Purchasing Cost**: Purchased Quantity * Purchasing Price
- **In House Cost**: Inhouse Price * In House Quantity
- **Outsourced Cost**: Outsourced Quantity * Outsourcing Price
- **By Product Cost**: By Product Quantity * By Product Price
- **Material Cost**: Purchasing Cost + Outsourced Cost + By Product Cost Total Cost: Operation Cost + Fix Cost + Material Cost

These costs are calculated only for the lines and not for the head.

** // Cost type 0-9 // **
The cost type which is referred by the calculation row.

1.5. Defining Price Lists

Before defining Price Schemas the user has to decide how to use the price lists in SAP Business One.

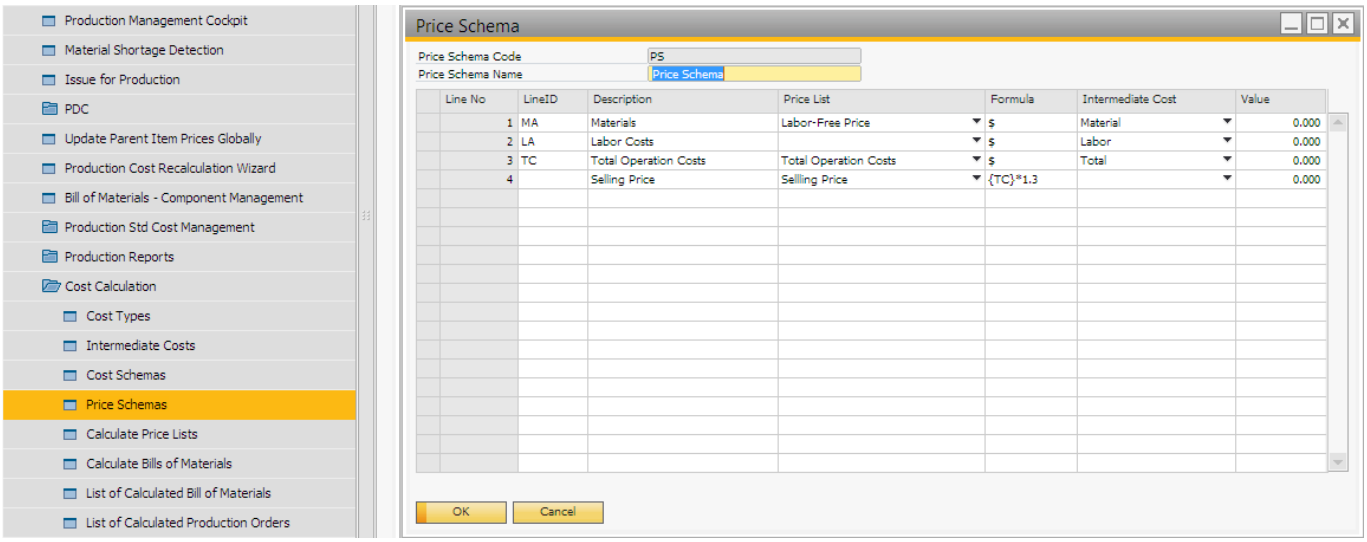


In our sample we have renamed the first three price lists. The price lists are referenced in price list

schemas.

1.6. Defining Price Schemas

The main goal of Price Schemas is to aggregate cost schemas intermediate results to final results, as well as to define a mapping between cost schemas and price lists. The values are taken from the cost schemas via the Intermediate Cost variables. The calculated values of the lines in a price schema may be linked to price lists.



Formula field values:

- $ = The ‘Intermediate Cost’ value.
- {LineID} = The result of the line with the given ‘Line ID’.
- [LineNo] = The result of the line with the given ‘Line No.’.
- [-x] = The result of the line which is x lines above.
- x% = A value in percent.

Example:

Line No	Line ID	Description	Price List	Formula	Intermediate Cost	Value
1	MA	Materials	Labor-free price	$	Material	0.000
2	LA	Labor Cost		$	Labor	0.000
3	TC	Total Operation Cost	Total Operation Costs	$	Total	0.000
4		Selling price	Selling price	{TC}*130%		0.000

In our sample the value of the Materials line is linked to the Labor-Free price. The formula field $ here refers to the selected Intermediate Cost. The Selling Price line is calculated as Total Operation Cost x 1.30.

It is possible to refine the formula by using the DataTableCompute method. For more information about the method please see:

- Expression
- Compute

Example:

Line No	Line ID	Description	Price List	Formula	Intermediate Cost	Value
1	MA	Materials	Labor-free price	$	Material	0.000
2	LA	Labor Cost		$	Labor	0.000
3	TC	Total Operation Cost	Total Operation Costs	$	Total	0.000
4		Selling price	Selling price	IIF({LA}=0, {MA}*150%, {TC}*130%)		0.000

In the second example the Selling Price line is only calculated as the Total Operation Cost multiplied by 1.3 if the Labor Cost is greater than zero. Otherwise the Selling Price is the Material Cost multiplied by 1.5.

MSSQL TIP
If the desired formula cannot be defined with the DataTable.Compute method, you can use SQL syntax to define the formula as well. Please keep in mind that defining several formulas with SQL syntax can lead to slower performance due the higher number of the SQL queries that the system runs.

Please note: Defining the formula with SQL syntax is not yet supported on HANA. Always use the DataTable.Compute method to define the formula.

Example

Line No	Line ID	Description	Price List	Formula	Intermediate Cost	Value
1	MA	Materials	Labor-free price	$	Material	0.000
2	LA	Labor Cost		$	Labor	0.000
3	TC	Total Operation Cost	Total Operation Costs	$	Total	0.000
4		Selling price	Selling price	CASE WHEN {LA} > {MA}*2 THEN {LA}*1.5 WHEN {MA} > {LA}*2 THEN {MA}*1.5 ELSE {TC} END		0.000

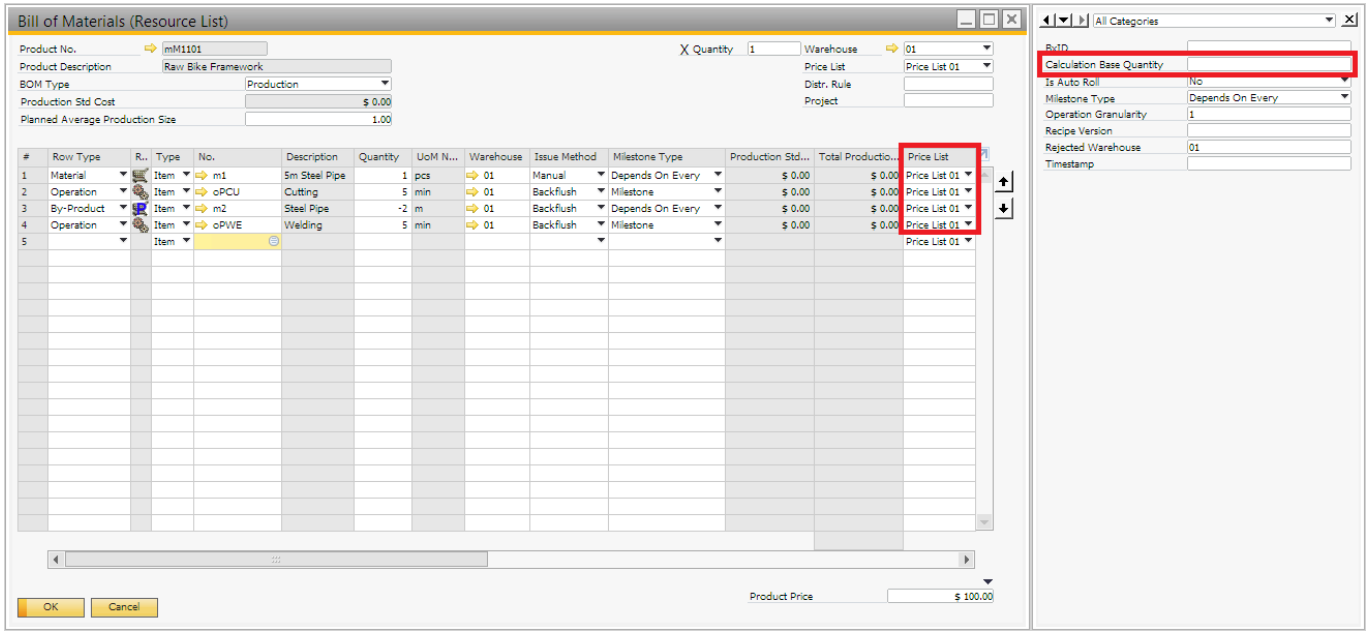
In this example the selling price is calculated with the following method: If the labor cost is greater than the double of the material cost, then the selling price is calculated by multiplying the labor cost by 1.5. If the material cost is greater than the double of the labor cost, then the selling price is the material cost multiplied by 1.5. Otherwise the selling price is the total operation cost.

2. Calculating Costs and Prices

At this point we have defined cost types for resources and calculation algorithms (schemas). The next step is to calculate production costs. Basically there are a number of possibilities for calculating the costs of a product. It is possible to calculate the costs of a product based on:

1. bill of material structure
2. the component list in a specific production order
3. the actual work and used materials reported via PDC

2.1. Reviewing Bill of Materials for Cost Calculations



The Price List in a BoM should be set to an unused price list, if the Produmex Manufacturing cost calculation module is intended to be used for calculating and updating price list prices for the product of the BoM. The reason is that whenever the Update button is pressed, SAP Business One automatically updates the price of the product for the price list defined.

The Price Lists in the component matrix are important; the calculation logic retrieves the prices for purchased material items from the price list defined in the BoM. The price list for operations and own-manufactured materials are calculated and not simply retrieved from price lists.

The Calculation Base Quantity is a estimated quantity of a typical production order. This number is

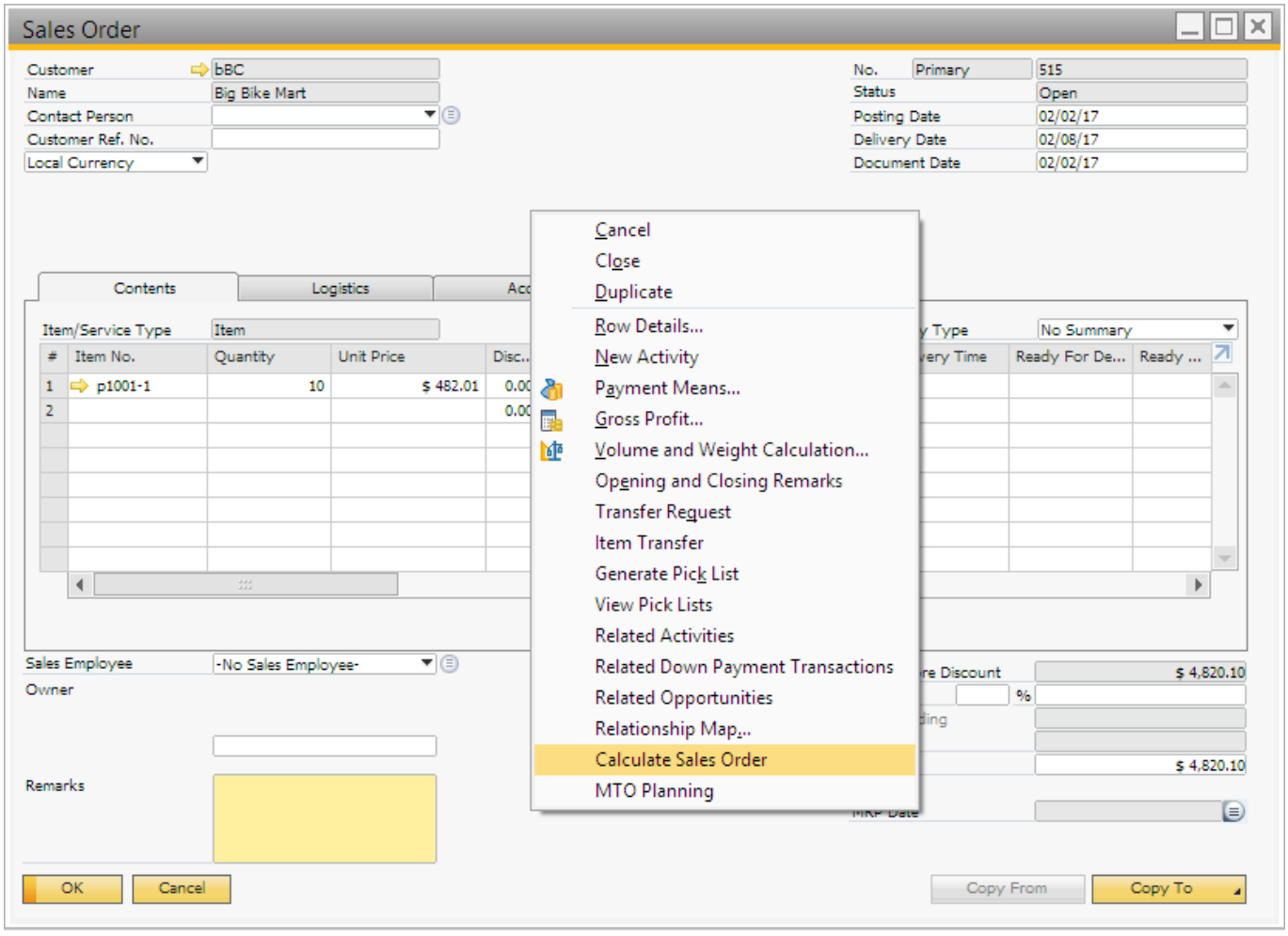
used when the setup and shutdown costs are calculated for a single unit of product.

2.2. Reviewing Item Procurement Methods

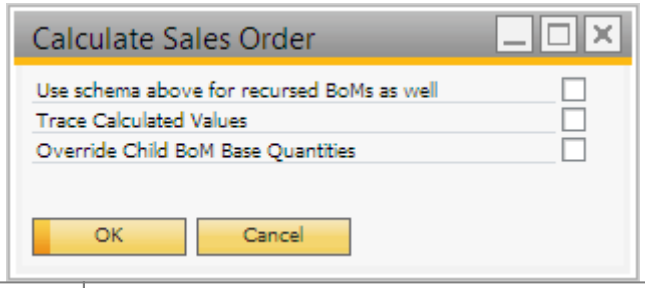
> It’s very important that the procurement method for own-manufactured components be set to “Make”; otherwise, the calculation engine will simply take its cost from a price list when the item is used as a component in another product’s BoM.

2.3. Sales Calculation: Calculating Sales Orders and Quotations

This is a preliminary calculation of costs before producing your product. You can start it from the sales order or the sales quotation with the right click menu:

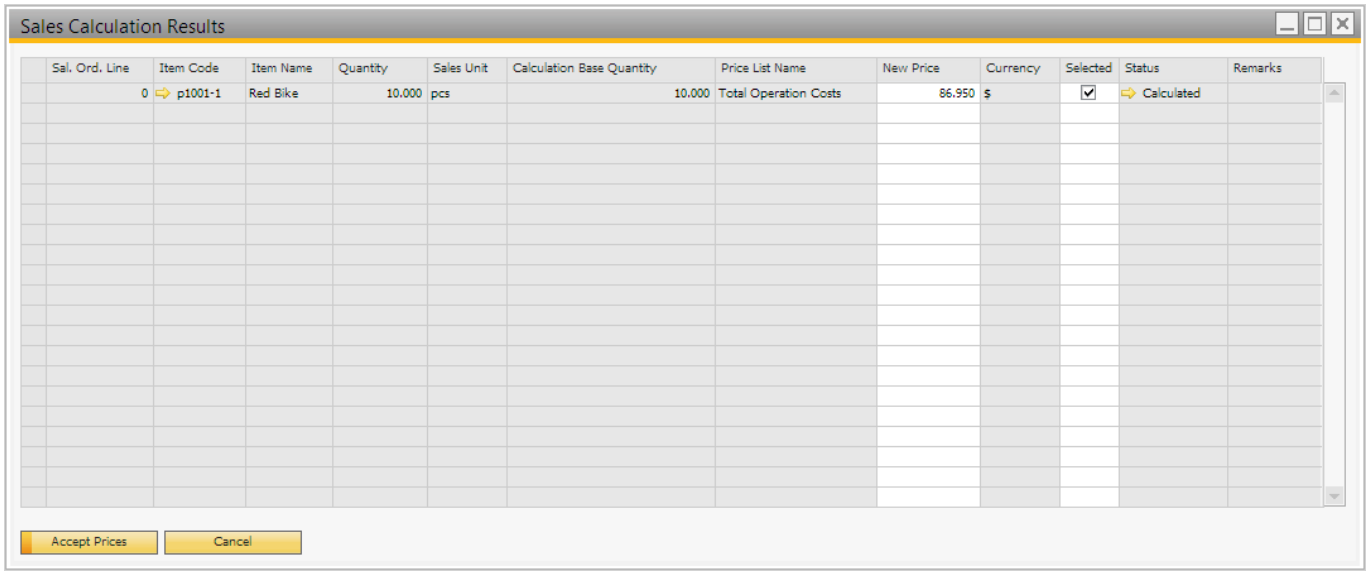


You will get a window with the calculation parameters:



Use schema above for recursed BoMs as well	The calculation schema of the main item will be used in all BoMs in the structure that are part of the main item.
Trace Calculated Values	There will be a golden arrow for calculated values. If you click on them, you will have a small explanation from where the value is coming (if available).
Override Child BoM Base Quantities	Base calculation quantity will be used in all child BoMs as well.

If you click on OK, the calculation will be done for all make items, and you will get a summary window with the calculation results.



You will see a list of all sales order/quotation rows and the items with the ordered/quoted quantity and the calculation base quantity from the BoM of the item. The calculation will take the higher from the Quantity and the Calculation Base Quantity and will use it as calculation base quantity. If the item is purchase item, it will show up in the list, but in the remarks field, you will see a message that is has not been calculated.

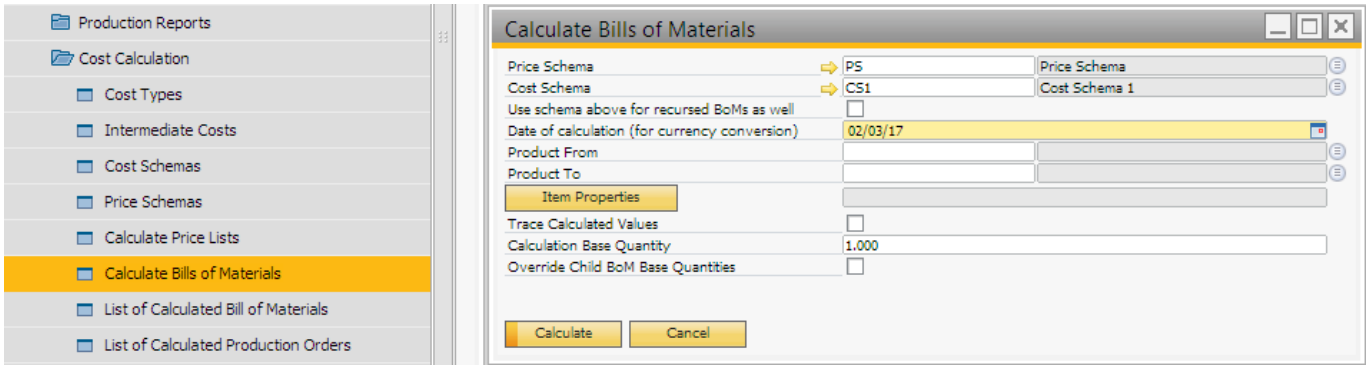
In the price list field you will see the price list that belongs to the item and the customer in the sales order/quotation. The New Price field contains the result of the calculation. It is possible to update the sales order/sales quotation with the calculated price. Check the 'Selected' checkbox on the line of the item(s) and click on the Accept Prices button.

To check the calculation details click on the arrow in the Status column. The arrow will not open up the result form if the calculation finished with an error or if the item is a purchase item.

During the sales order calculation, currency differences are not taken into consideration.

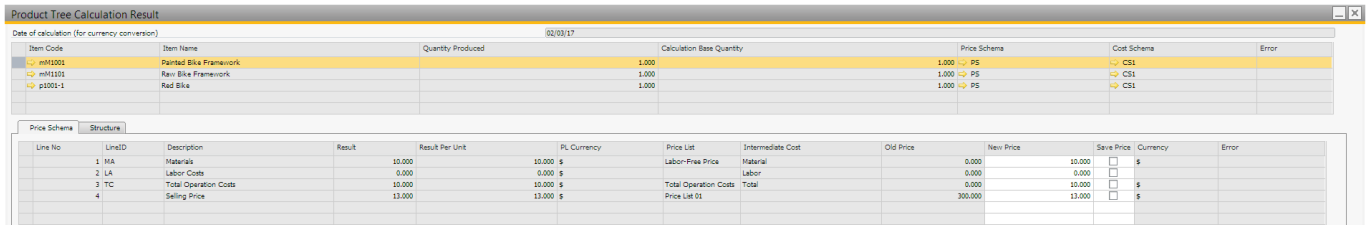
2.4. Calculating Costs from Bill of Materials

The most straightforward and simple way of cost calculation is when the cost of our products are calculated based on their, usually hierarchical, component structure in their BoMs. In the following sample we calculate the costs of all our products using the cost and price schemas shown above.



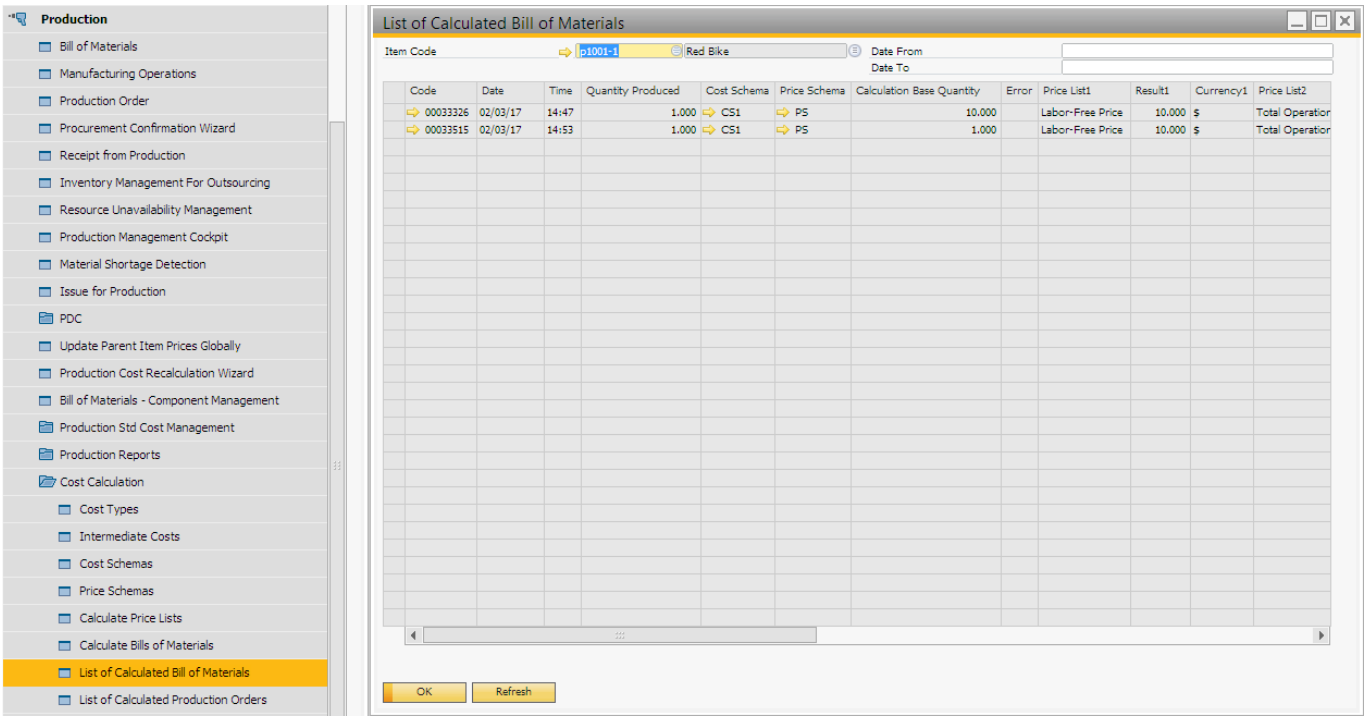
The main grid of the results form contains all our products that have BoM.

This form requires bigger screen resolution than 1024×768 to avoid that the OK button should not overlap the bottom matrix.



An important thing with the way the costs are calculated that the intermediate costs are collected hierarchically from subordinate components as if the primary had a giant BoM. For example, the Labor Costs calculated for the product p1001-1 (Red Bike) contains the labor costs calculated for the subordinate component mM1001 (Raw Bike Framework). The item mM1001 is a material component in the BoM of p1001-1. Since mM1001has its own BoM and its procurement method is set to “Make”, it is not calculated as a simple material with a price list price. With this way the material cost, for example, of p1001-1 is the sum of all the cost of purchased materials (procurement method = Buy) in the BoM tree.

//Note: in newer versions of Produmex Manufacturing all calculation results are saved, and you can review them later on. To do this open the List of Calculated Bills of Materials from the right click menu in the BoM. //

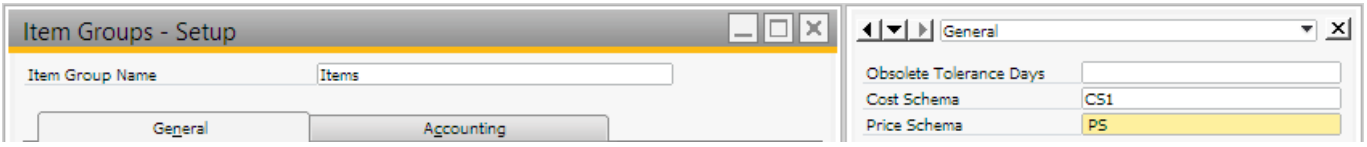


2.5. Price List Recalculations

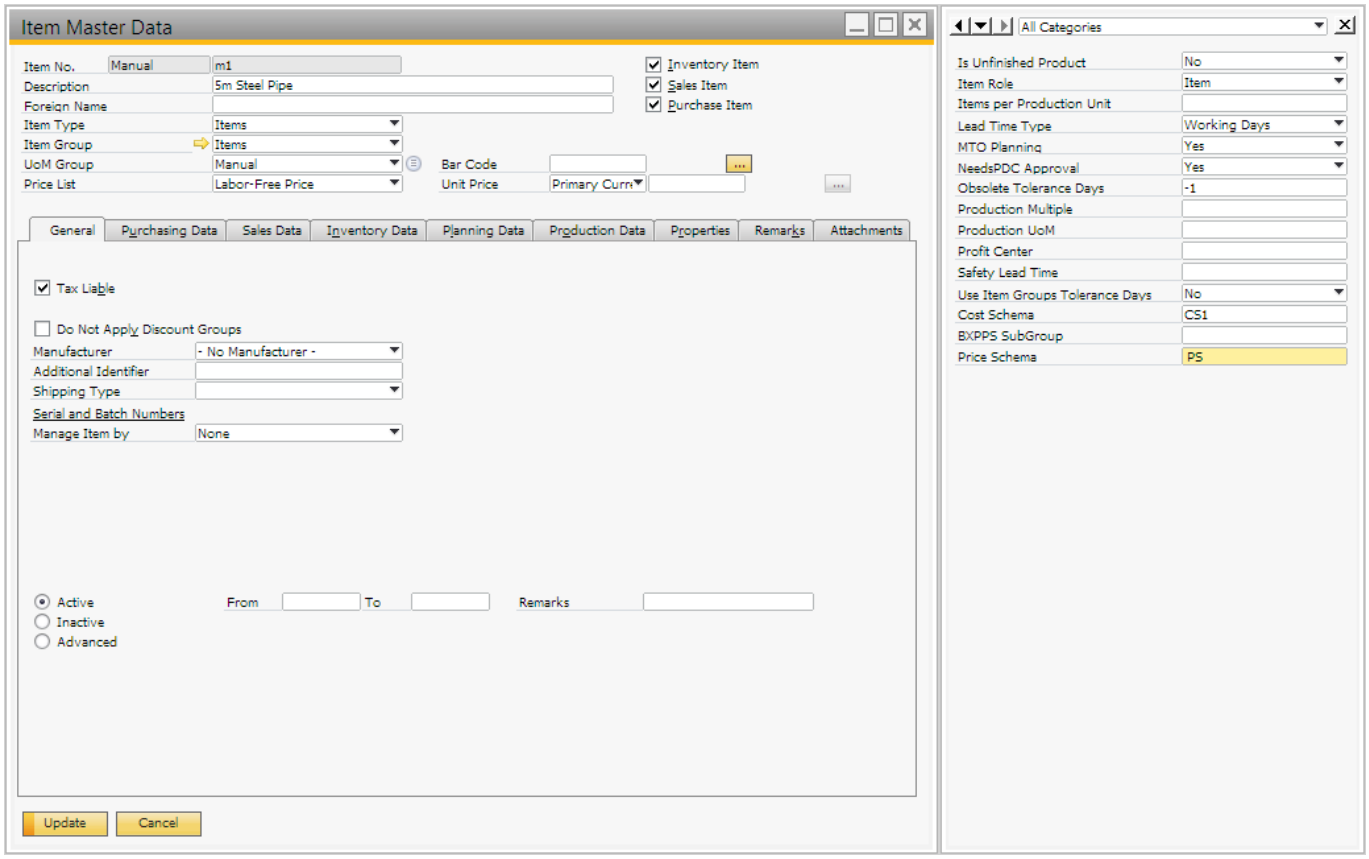
A main objective of the cost calculation module of Produmex Manufacturing is that the standard price lists in SAP Business One can be updated with the calculated prices with a single button click.

2.5.1. Defining Default Schemas for Items and Item Groups

Since Produmex Manufacturing allows the definition of multiple calculation schemas and each item may have different schemas, the simplest way to associate items with calculation schemas is when schemas are defined for the item groups.



The items inherit the schemas defined for their item group, but these can be overridden in the Item Master Data form.



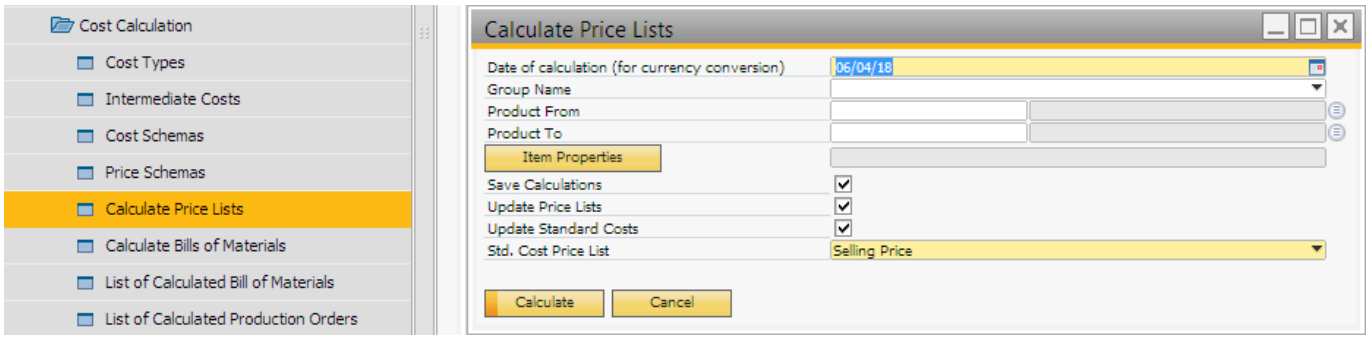
2.5.2. Batch Updating Price Lists with Calculated Prices

When the schemas are in place and all the manufactured products are associated with a price schema and a calculation schema, you can batch calculate the items and update the price lists based on the calculated values.

Select the ’Calculate Price Lists’ option. On the Opening Calculate Price Lists form define the date of the calculation. It is possible to narrow down the calculation to an item group, to selected items or to items with certain properties. Define the filters on the ’Group Name’ and ’Product From-To’ field or click on the Item Properties button to select the filtering item properties.

In order to save the results of the calculation in the database, enable the ’Save Calculations’ button.

Press the ‘Calculate’ button to calculate the price lists.

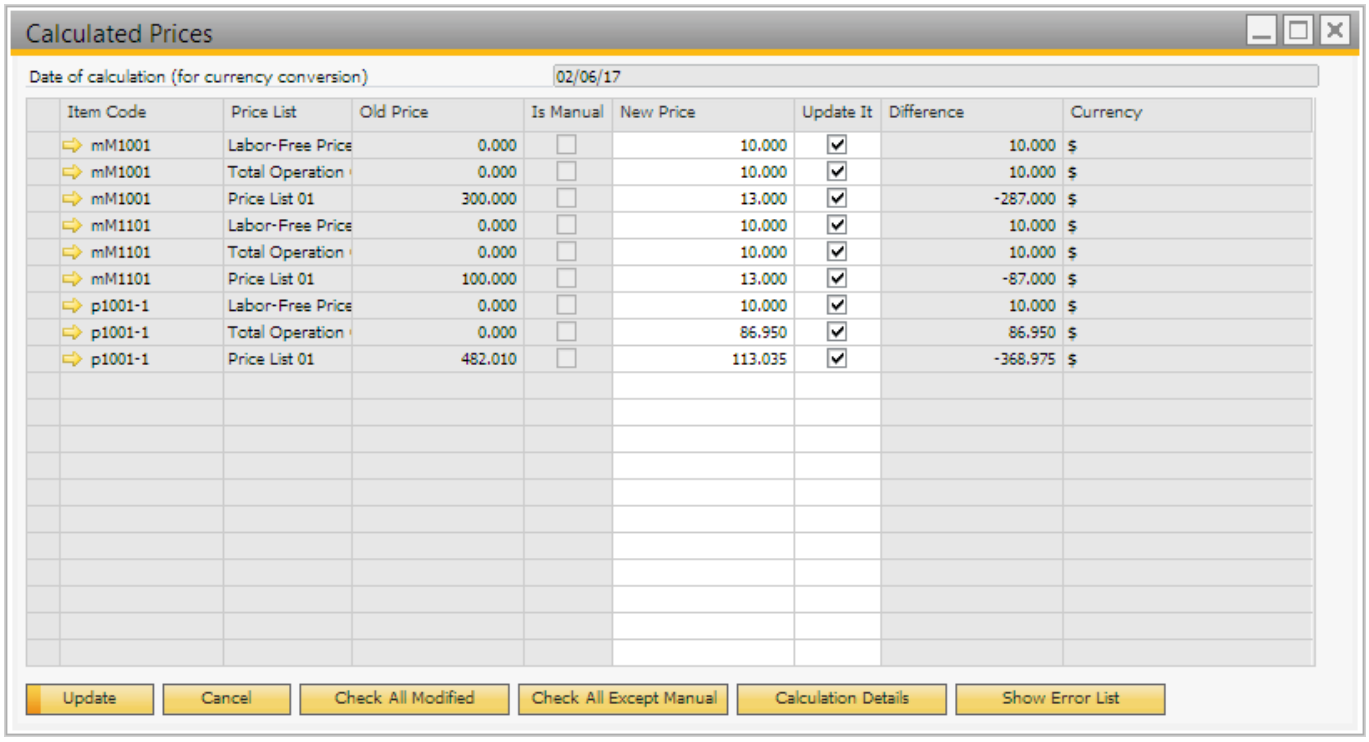


When the calculation process is completed, the Calculated Prices form is opened. On this screen every active inventory item that has a Bill of Materials is listed. Values for different price lists are displayed on separate lines.

By default the ‘Update It’ checkbox is checked if there is a difference between the old and the new price for the item in the given price lists. Press the ‘Update’ button to update the prices to the new prices on every line where the ‘Update It’ checkbox is checked.

If the ‘Update Price Lists’ checkbox was enabled on the Calculate Price Lists form, the prices are automatically updated after the calculation.

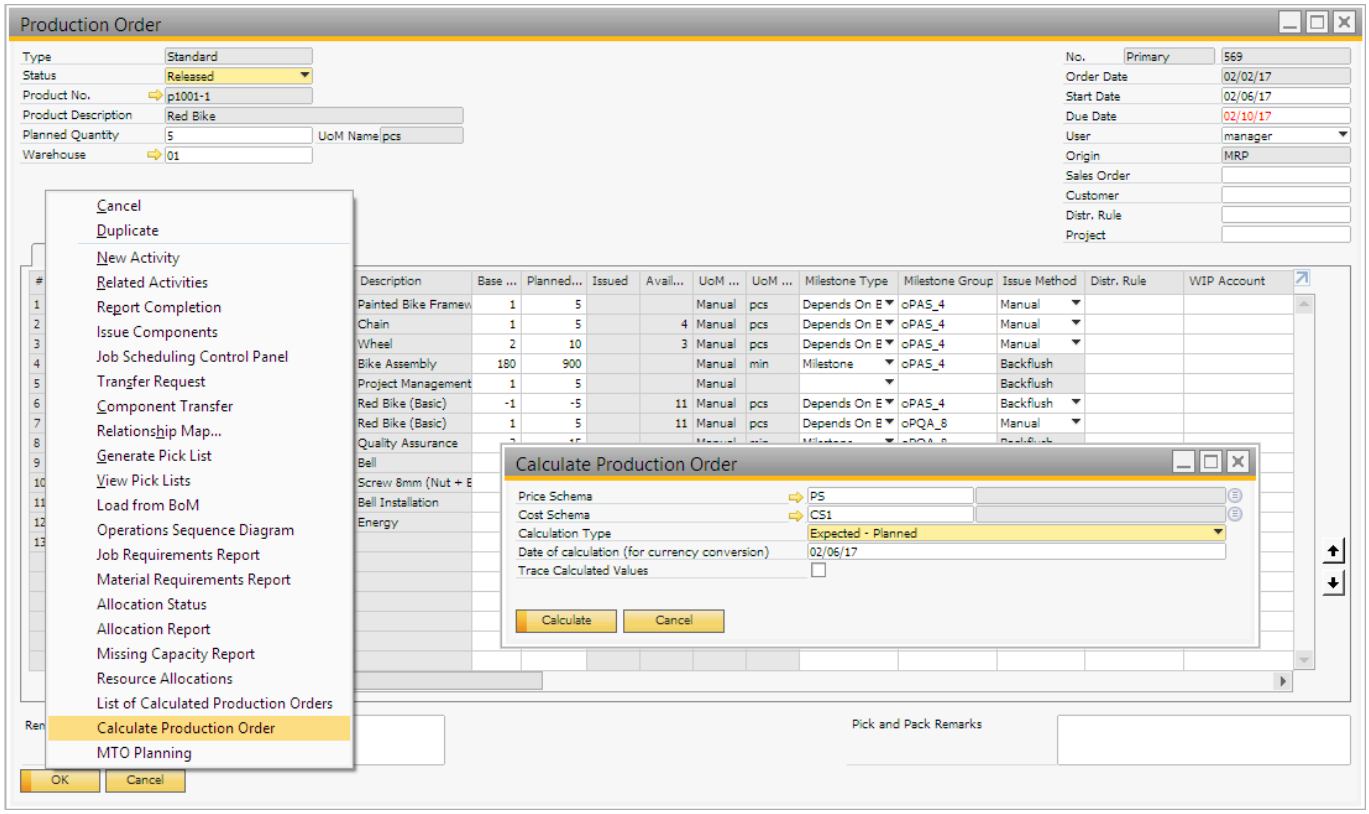
It is also possible to update the standard cost for the items. If the ‘Update Price Lists’ checkbox is enabled, the ‘Update Standard Costs’ checkbox becomes active. In order to update the standard cost as well, check this checkbox and select the source price list from the ‘Std. Cost Price List’ dropdown menu.



Pressing the Calculation Details button will open Product Tree Calculation Results form containing the details of calculations for all the items.

2.6. Calculating Production Orders

When a right-click menu is opened on Production Order form, the user may select the Calculate Production Order menu.



In the parameter form the user selects the Price and Cost Schema for the calculation and the source of the (planned) resource consumption/allocation of the production order. Note: All calculation results are saved just like the BoM calculations, so that you can compare them later on. You can find the saved calculations in the right click menu List of Calculated Production Orders. Some generic information about the calculation method of production order calculation:

- In the production order calculation the structure of calculation is flat. The BoMs of materials (if there is any) is not expanded. Second level rows are for resource allocations under the operations. They contain the real cost of the operation as different work centers can be assigned to the same operation with different costs.
- In planned and released mode the produced quantity is the planned quantity of the product.
- In actual mode the produced quantity is the completed quantity of the production order.
- In planned mode the cycle count is always one, the setup and teardown time is multiplied by the cycle count.
- In released mode the cycle count is the number of allocations and the setup and teardown time is multiplied by it.
- In actual mode the cycle count is the number of start setup bookings created for that operation and the setup and teardown time is multiplied by it.
- The purchase price for the materials comes from the SAP item cost in the item master data except for the actual mode where it comes from the issue for production bookings.

2.6.1. Planned Cost Calculation for Production Orders

In the case of “Expected – Planned”, the source of resource usage is the component list of the production order. When a “standard” production order is first created the component list is copied from the BoM of the item. This component list can be modified for a production order; therefore, the component list with its quantities could be significantly different from the original BoM of the product. When the production order is in “Planned” status the only meaningful calculation type is “Expected –

Planned". Note, that when a production order is in planned mode, no actual work centers are allocated (unless mandatory work centers are manually defined for the operations). Because of the logic, at this time the cost amounts may come from work center features and operations but not from actual work centers.

2.6.2. Released Cost Calculation for Production Orders

When a production order is released, Produmex Manufacturing allocates actual work centers for the operations. If the cost types are more specifically defined for work centers the "Expected – Released" calculation type may be more specific since in this case the cost amounts are coming from the work centers (if they are defined to override the cost amounts from higher levels). Even when no cost amounts are defined for work centers the operation cost for released production orders may be slightly higher because of the multiple setup and shutdown costs of the operations. When a production order is released the required resource capacities are allocated and reserved. During this resource allocation multiple work-centers may be allocated for an operation and if that operation has setup and shutdown costs the operation cost will be slightly higher than calculated for a planned production order.

2.6.3. Actual Cost Calculation for Production Orders

With the Actual calculation type, a precise idea of the cost of every job can be obtained. With these data a commercial analysis of the production process can be carried out. The quantities of materials come from the Issue for Production transactions. The costs of operations come from PDC. The prices for materials, normally, come from the Inventory Master Data (OITM.AvgPrice or OITW.AvgPrice). Remember that in Production Orders there is no possibility to define price lists for the material components. The prices of batch and serial numbered components can come from the same source as the normal components.

From:
https://wiki.produmex.name/ - **Produmex**

Permanent link:
https://wiki.produmex.name/doku.php?id=implementation:manufacturing:costcalc

Last update: **2021/04/20 09:20**