MOPP XXI Tutorial

1 Previous work

Due to the enormous complexity of a multipurpose batch chemical plant it is necessary to define the case study on paper work before start to input the data in MOPP. A general idea about the results that should be obtained it is also necessary.
The following aspects should be known:

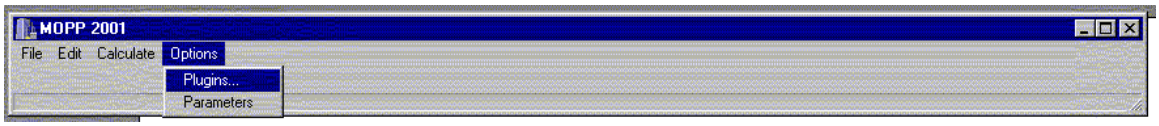
- Number of processing units and their capacities
- Number of products, intermediate materials and final products to take into account and their capacities.
- Number and characteristics of other resources available (manpower, electricity, steam, etc.)
- Recipes to be defined and their associated constraints and resource consumption
- Operation times
- Storages available

In our example there are 5 units available producing three products (A,B and C). Three is one resource involved (electricity) and two storages dedicated to materials A and B.

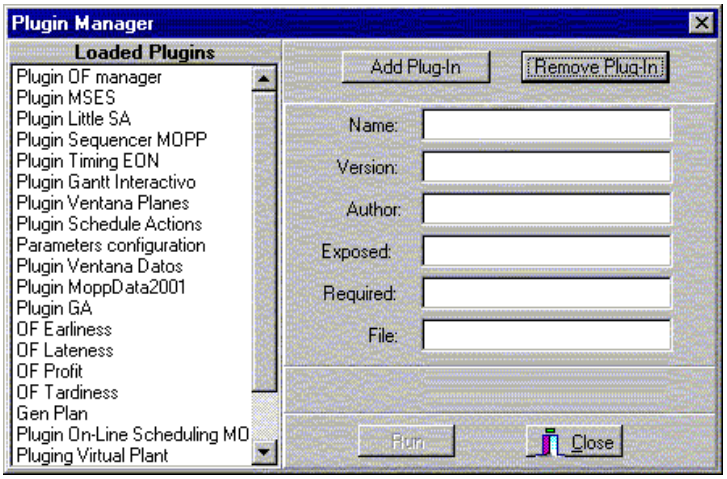
Paper templates are available in order to organize the previous work more easily.

2 Starting to work with MOPP

MOPP framework is plugin-based. It means that all the MOPP functionality's are carried out by means of plugin connected to MOPP.
The first step is to assure that the necessary plugins are present. To check it we have to open the main mopp program and check the plugin configuration:



The plugin manager window should appear showing the current active plugins..



In order to follow the tutorial the following plugin should be loaded into the system:

resourceview.dll MOPP XXI data manager/editor

The plugins loaded when MOPP XXI is started are listed in the configuration file config.plg. As a plain text it can be edited manually using notepad or another plain text editor.

3 Problem data input

3.1 The resourceview.dll plugin

The resourceview.dll plugin provides a data editing interactive window. When connected to MOPP this window sets and additional Menu options to the main menu:

File -> New	Clears the actual information from memory
File-> Load MOPP files	Load a file with .mopp extension
File-> Save MOPP files	Save a file with .mopp extension
File->Load old XML files	Load a file with .xml extension (schedules are not loaded)
File->Save old XML files	Saves a file with .xml extension (schedules are not saved)
File->Load from Database	Load a new case from relational Database
File->Save to Database	Saves a new case to relational database
Edit->MoppData	Runs the data input window.

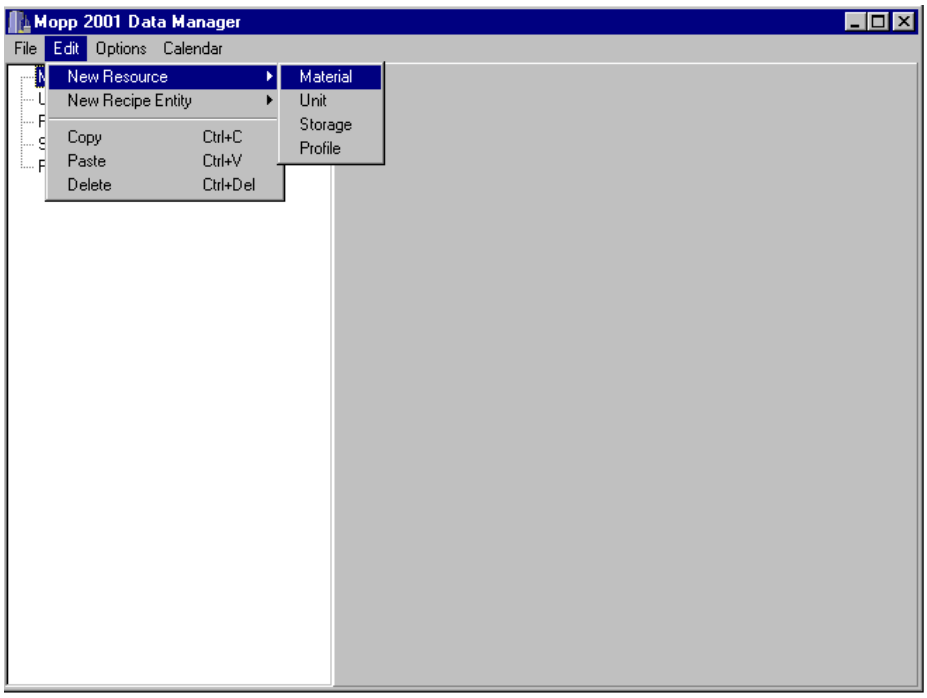
This plugin interacts with moppdata.dll plugin which is the plugin that really stores all the information contained in MOPP.

3.2 Materials

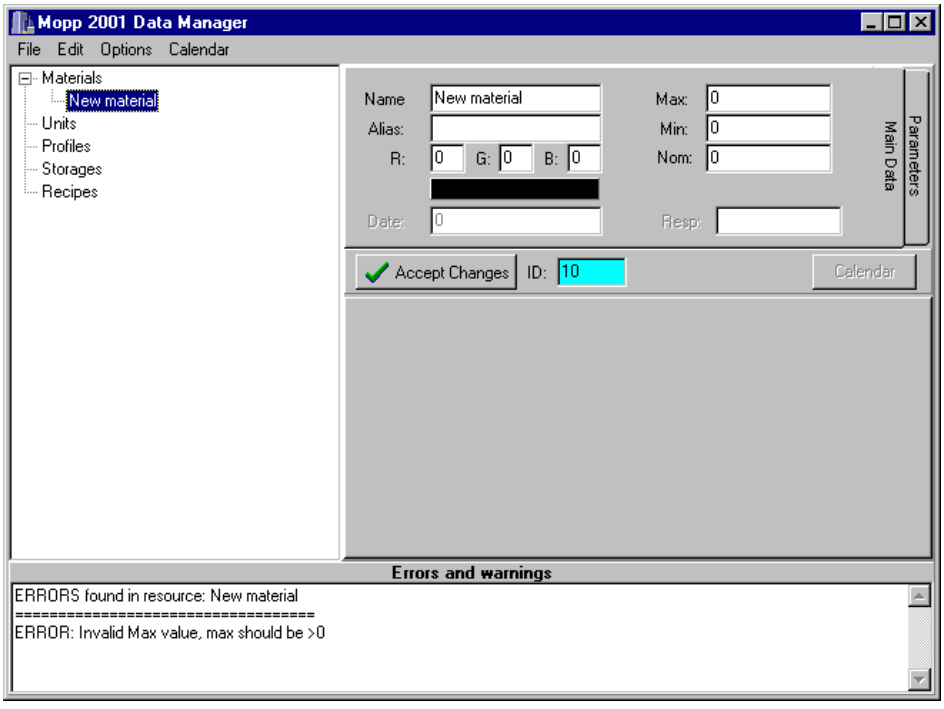
The first step to define a new case is to define the materials involved. In our test case the following materials should be defined:

MATERIALS			
Name	Max.	Min.	Nom.
Material A	10000	0	0
Material B	10000	0	0
Material C	10000	0	0

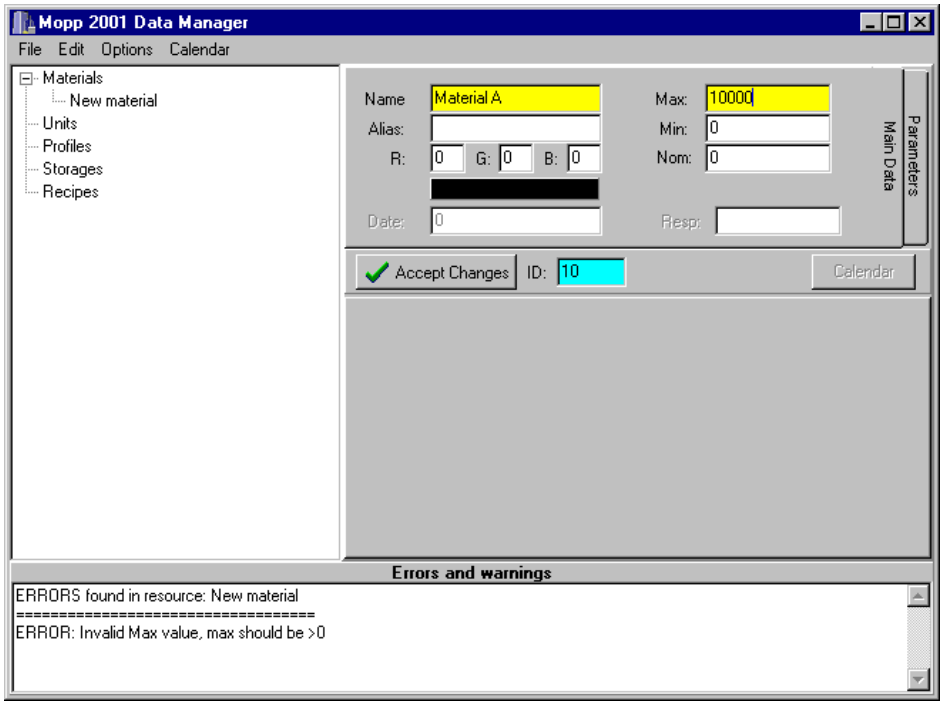
The first step to introduce this information in MOPP is to open the moppData window. This action if perform with the menu option Edit->MoppData. The result of this action if the following window:



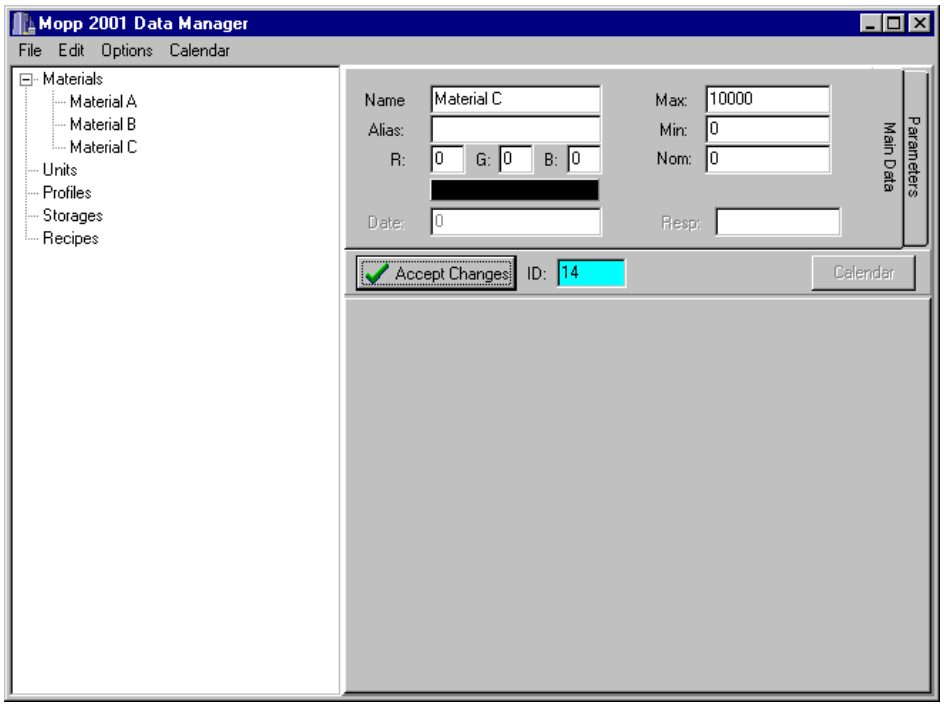
The menu option Edit->NewResource->Material allow to create new materials. When using this option the aspect of the window changes allowing the edition of the new material as it is show in the next picture:



The default values should be changed to match the values of each of the materials defined as it is shown in the following picture. The button “Accept Changes” should be used in order to store the modifications of the resource in the memory.



Using the copy & paste (Ctrl-C and Ctrl+V) capacity, the remaining materials can be easily introduced. Resulting on the following aspect of the interface:

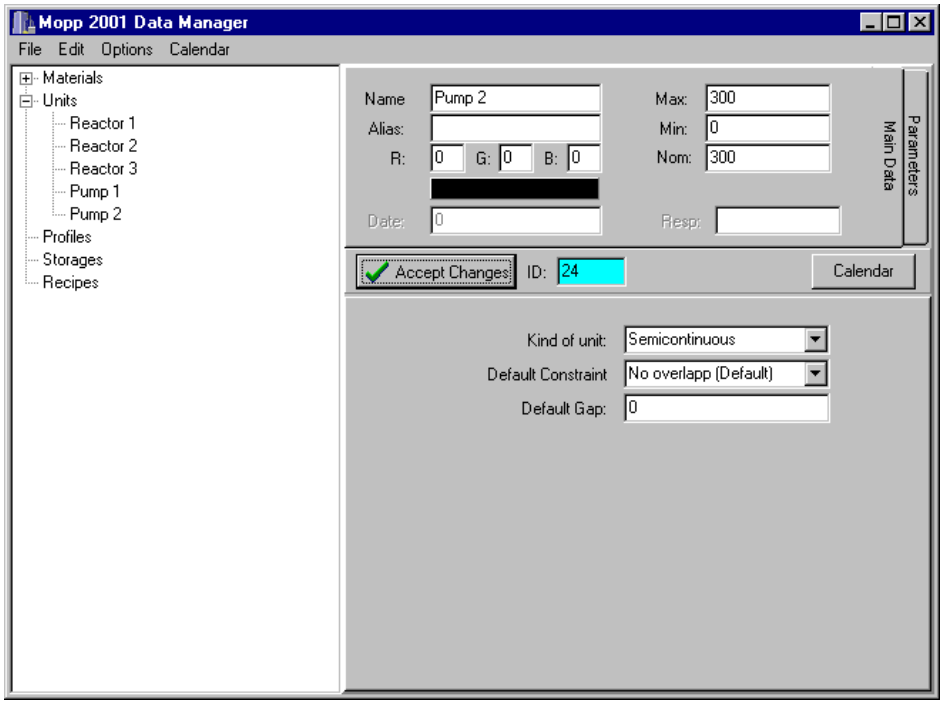


3.3 Units

The next step of defining the tutorial case is to introduce the processing units involved in the process. As in the previous case, a new unit can be created using the corresponding menu option (Edit->New Resource->Unit)
In the tutorial case, the data to be introduced is the following one:

UNITS				
Name	Max:	Min	Nom	Kind
Reactor 1	1000	0	1000	Discontinuous
Reactor 2	700	0	700	Discontinuous
Reactor 3	700	0	700	Discontinuous
Pump 1	200	0	200	Semicontinous
Pump 2	300	0	300	Semicontinous

Once introduced, the interface should show this aspect:



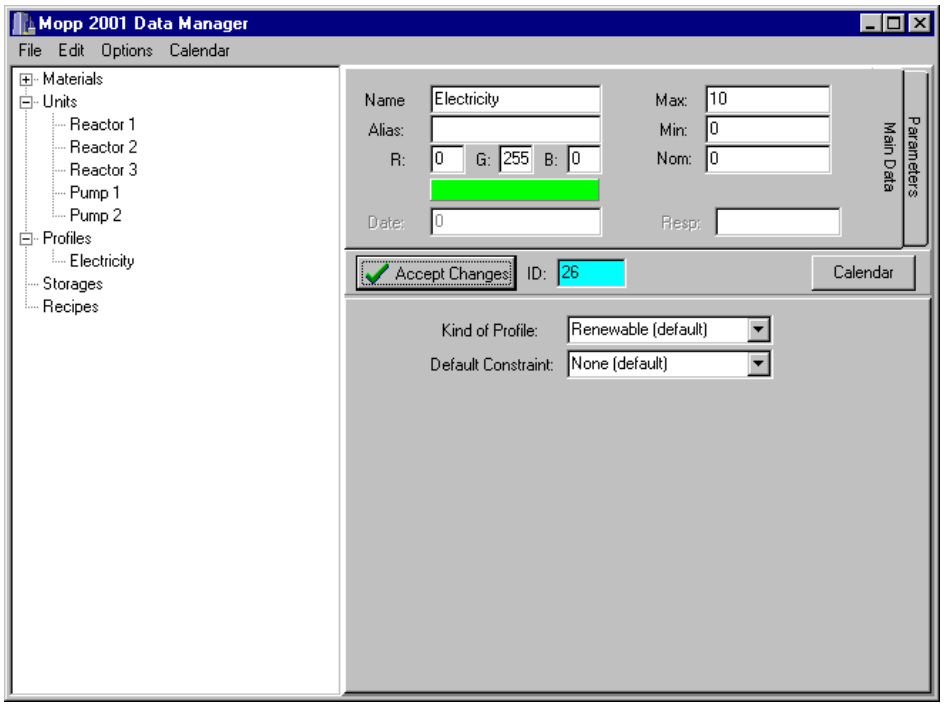
The copy & paste functionality can also be used in order to create the different units in a fastest way. It is very important to put a value different than 0 in the Nominal capacity of the unit. This value is used in order to calculate the Batch size of a batch that uses that unit.

3.4 Profiles

The next information to introduce is the information regarding the profiles. In This case only one profile will be defined: Electricity consumption. The data regarding this resource in detailed in the following table:

PROFILES				
Name	Kind	Max.	Min.	Constraint
Electricity	Renewable	10	0	None

A new profile can be introduced using the menu option Edit->New Resource->Profile. The resulting aspect of the interface is shown in the following figure.



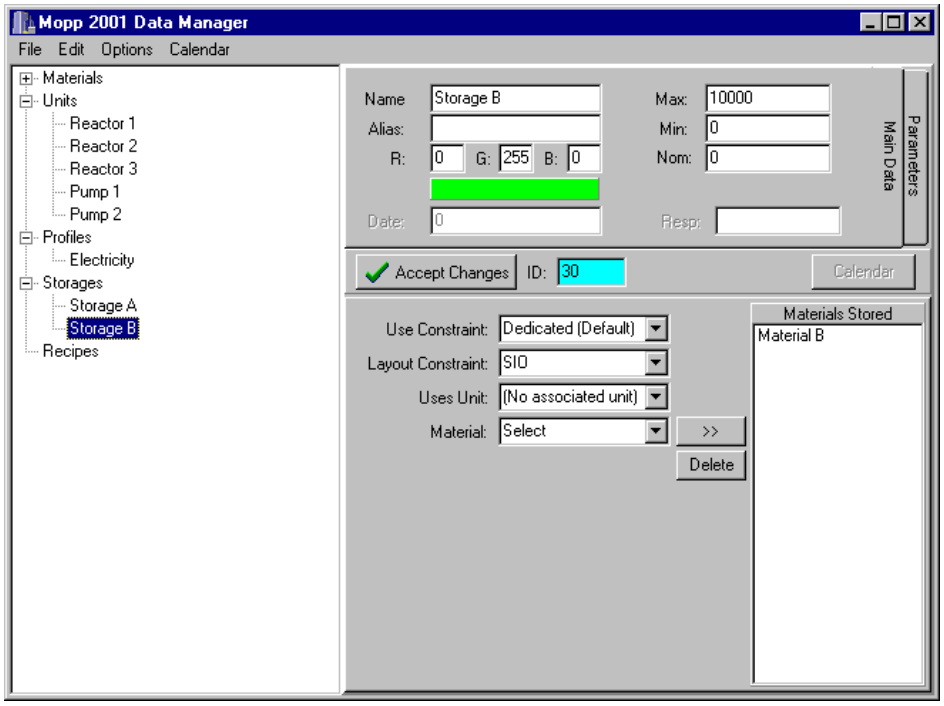
It is convenient to define a specific color associated to the profile, as it will be the color used to represent it in the gantt chart.

3.5 Storages

The last plant resources to define are the storages. There is the corresponding menu option to create a new storage (Edit->New Resource->Storage). The tutorial case study has two storages associated as they are specified in the following table:

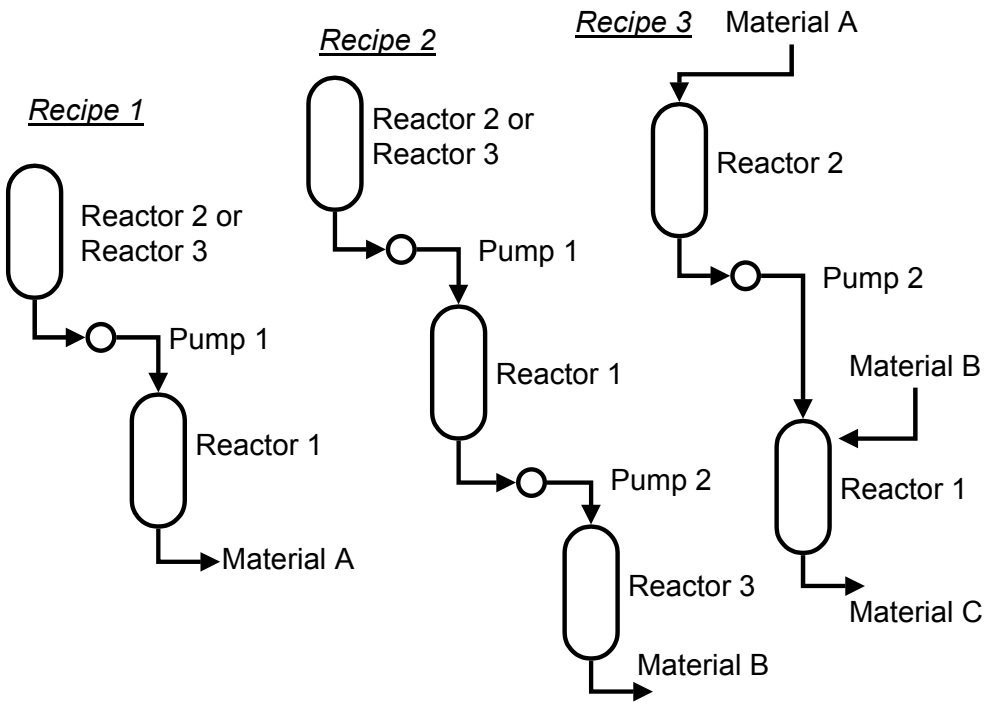
STORAGES				
Name	Max	Use constraint	Layout constraint	Materials associated
Storage A	10000	Dedicated	SIO	Material A
Storage B	10000	Dedicated	SIO	Material B

The methodology used to fill the different properties of the storage is the same as in the other resources. As in the case of the profiles, it is convenient to assign a color different than black to each storage. This color will be used to represent the occupation of the storage in the gantt chart. .The resulting aspect of the interface with all the storages introduced is shown in the following picture.



3.6 Recipe Description

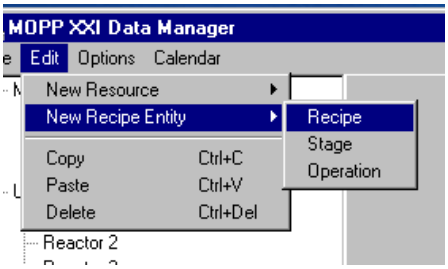
This case study has three different recipes shown schematically in the following figure:



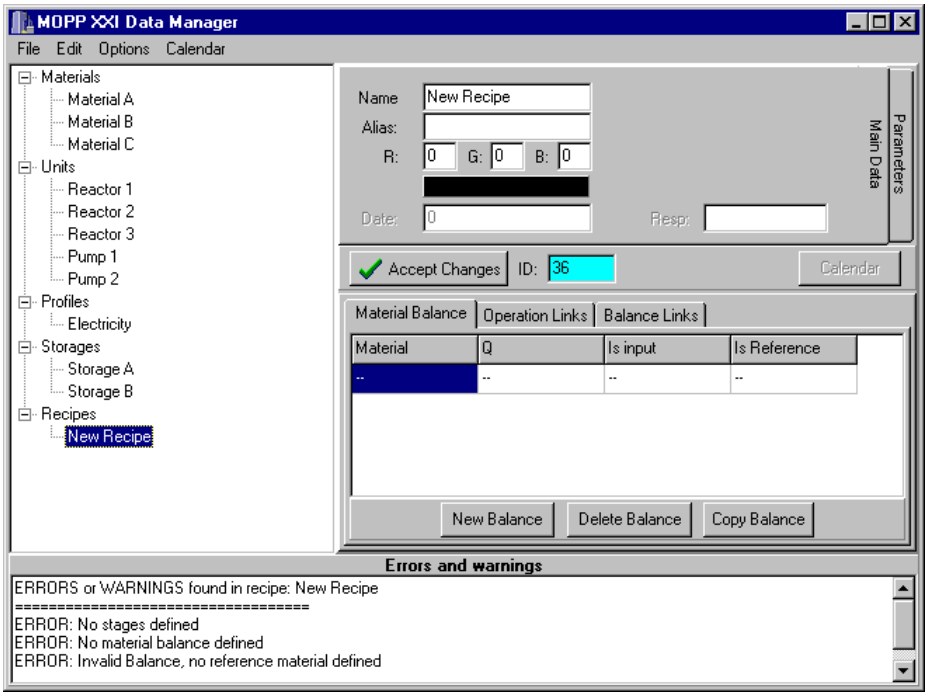
There are three different recipes which produces material A,B and C. In order to produce material C, materials A and B are needed. It can be observed that Reactor 1 is shared between the three recipes, and Reactor 3 can be reused in Recipe 2.

3.7 Creating a Recipe

The recipes can be created using the option Edit->New Recipe Entity->Recipe as it is shown in the next figure:



The aspect of the window with a new recipe created is the following one



There are three tags present at the Recipe level:

- Material Balance: For description of the global material balance of the recipe
- Operation Links: For description of the time constraints between operations of the recipe
- Balance Links: To establish the real flows referenced to the global material balance assigned to the different operations.

The first step on introducing a new recipe is defining the global material balance. One material should be established as Reference Material (Column Is Reference to **Yes**). The batch size of a batch using this recipe

will be referred to the Reference Material. Fort example lets imagine that we have a recipe with the following material balance relation:

$$2A+1{,}5B \rightarrow C$$

This means that 2 units of A and 1,5 of B is needed to produce 1 unit of C. If the reference material is C, a Batch with a batch size of 30 will create the following material balance:

- Inputs
  - 60 units of A (30*2)
  - 45 units of B (30*1,5)
- Outputs
  - 30 units of C (reference material)

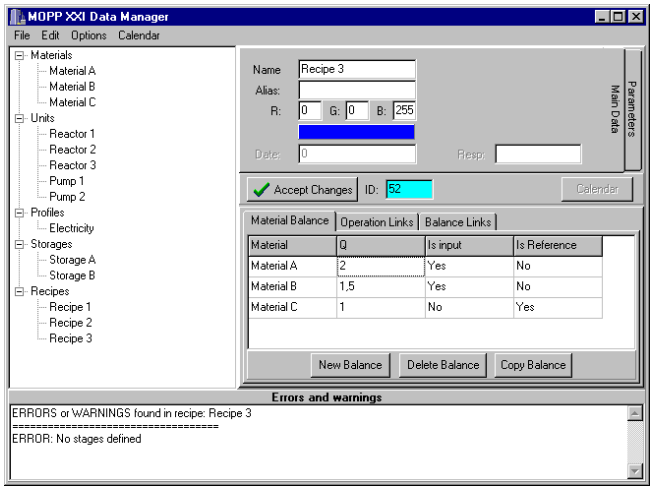
But if the reference material is B, the same recipe with a batch size of 30 will produce the following material balance:

- Inputs
  - 40 units of A (30*2/1,5)
  - 30 units of B (reference material)
- Outputs
  - 20 units of C (30/1,5)

In our case there are the following material balances to be defined in the three recipes

Material Balances				
Recipe	**Material**	**Q**	**Is Input**	**Is Reference**
Recipe A	A	1	No	Yes
Recipe B	B	1	No	Yes
Recipe C	A	2	Yes	No
	B	1,5	Yes	No
	C	1	No	Yes

To create new lines of the balance use the *"New Balance"* button. Once introduced, the interface shows this aspect:



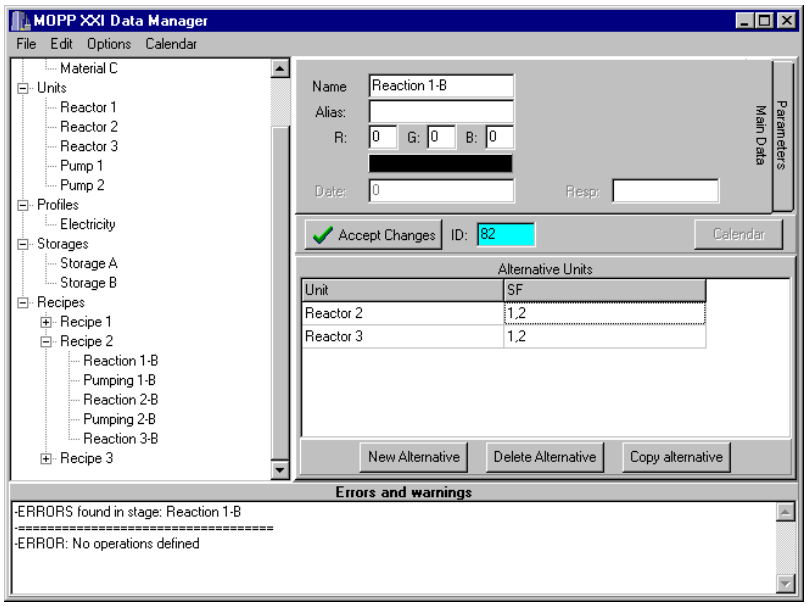
The next step is defining the stages present in the recipe.

3.8 Stages

One stages is defined as an ordered set of operations. At the level of stage is defined the possible alternative units that can execute that stage. This is the main information concerning stages. In our case the stages and alternative units to be introduced (option Edit->New Recipe Entity->Stage) are the following ones:

STAGES		
Recipe 1		
Stage	**Size Factor**	**Unit**
Reaction 1-A	0,8	Reactor 2
	0,8	Reactor 3
Pumping	1	Pump 1
Reaction 2-A	1	Reactor 1
Recipe 2		
Stage	**Size Factor**	**Unit**
Reaction 1-B	1,2	Reactor 2
	1,2	Reactor 3
Pumping 1-B	1,2	Pump 1
Reaction 2-B	0,9	Reactor 1
Pumping 2-B	0,9	Pump 2
Reaction 3-B	1	Reactor 3
Recipe 3		
Stage	**Size Factor**	**Unit**
Reaction 1-C	0,7	Reactor 2
Pumping 1-C	0,7	Pump 2
Reaction 2-C	1,2	Reactor 1

Once introduced, the graphical user interfaces shows this aspect:



The size factor is used to calculate the maximum, minimum and nominal Batch size of a recipe according the following criteria.
Given
$Umax_j$, $Umin_j$ and $Unom_j$: Maximum, minimum and nominal capacity of the unit assigned to stage j
SF_j : Size factor of the stage j using the unit with the capacities above.
Then maximum, minimum and nominal the batch sizes (BSmax, BSmin and BSnom) are calculated as:

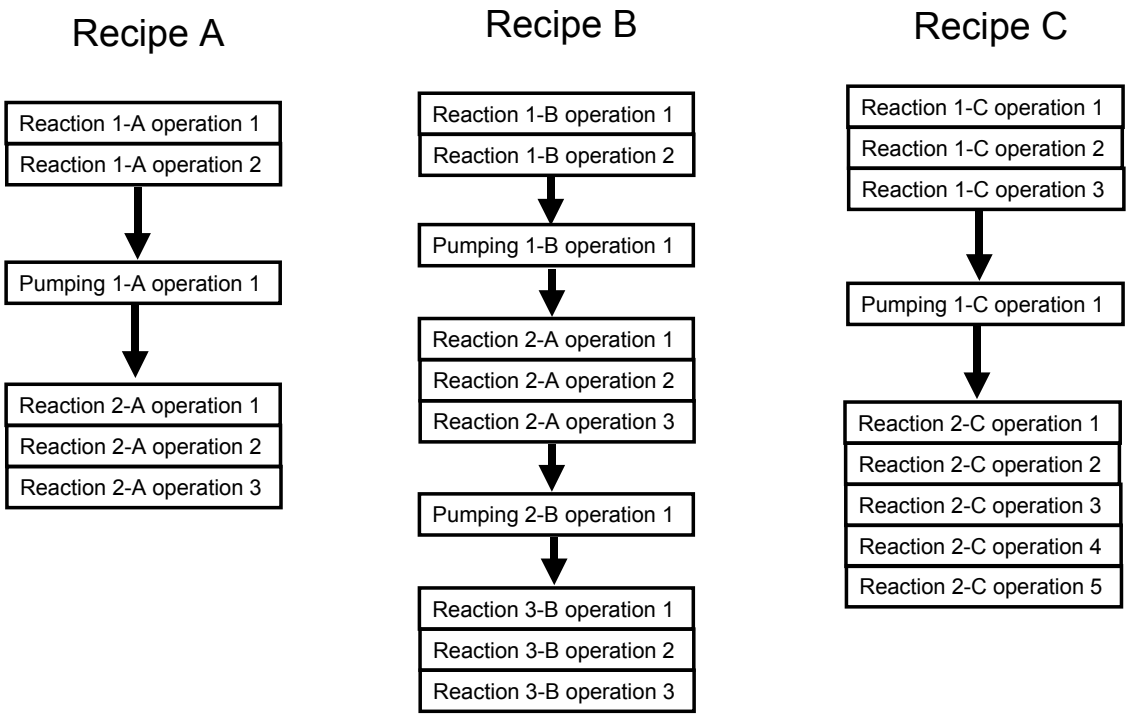
$$BS\max = \min\left(\frac{U\max_j}{SF_j}\forall j\right)$$

$$BS\min = \max\left(\frac{U\min_j}{SF_j}\forall j\right)$$

$$BSnom = \max\left(BS\min, \min\left(BS\max, \frac{Unom_j}{SF_j}\forall j\right)\right)$$

3.9 Operations

The last level of description is the operations level. Operation is the highest level of detail in the recipe description.
This detail can be schematically represented as follows:



The operations are created as the recipes and stages (option Edit->New Recipe Entity->Operation). Each operation have information concerning the time required to perform that operation (Time Data tab) and the Possible consumption of resources (Resources Data tab).
The time data is defined by 4 values (A,B,C and TW_{MAX}) the operation time (TOP) is calculated by default using the formula:

$$TOP = A + B \cdot Q^C$$

Where A,B and C are the values entered in the user interface and Q is the batch size processed in a given batch.

TW_{MAX} is used as a possible gap for the real operation time. Using all the parameters, and if the start time of the operation is *ST* and the end time of the operation is *ET*, then the calculated time for the operation must accomplish this constraint:

$$A + B \cdot Q^C \leq ET - ST \leq A + B \cdot Q^C + TW_{MAX}$$

if TW_{MAX} =0 then the constraint is simplified

$$ET - ST = A + B \cdot Q^C$$

And if B=0 and TW_{MAX}=0 then, it is even much more simple

$$ET - ST = A$$

So the use of these 4 parameters allows the specification of the time needed to perform an operation in a very flexible way.

If the unit used in an operation is of kind Semicontinuos and there is not a definition of time for this operation, then the operation time is calculated according the nominal capacity of the unit according this formula:

$$TOP = \frac{BS \cdot SF}{U_{NOM}}$$

The operations and the corresponding time data of the tutorial case are the following ones:

Operations Recipe 1					
Stage	**Operation**	**A**	**B**	**C**	**TW_{max}**
Reaction 1A	Operation 1	6	0	1	0
Reaction 1A	Operation 2	0	0	1	1000
Pumping 1A	Operation 1	*No time data introduced*			
Reaction 2A	Operation 1	0	0	1	1000
Reaction 2A	Operation 2	3	0	1	0
Reaction 2A	Operation 3	0	0.0005	1	0

Operations Recipe 2					
Stage	**Operation**	**A**	**B**	**C**	**TW_{max}**
Reaction 1B	Operation 1	4	0.003	1	0
Reaction 1B	Operation 2	0	0	1	1000
Pumping 1B	Operation 1	*No time data introduced*			
Reaction 2B	Operation 1	0	0	1	1000
Reaction 2B	Operation 2	0	0.005	1	0
Reaction 2B	Operation 3	0	0	1	1000
Pumping 2B	Operation 1	*No time data introduced*			
Reaction 3B	Operation 1	0	0	1	1000
Reaction 3B	Operation 2	4.32	0	1	0
Reaction 3B	Operation 3	0	0.0005	1	0

Operations Recipe 3					
Stage	**Operation**	**A**	**B**	**C**	**TW_{max}**
Reaction 1C	Operation 1	0	0.0005	1	0
Reaction 1C	Operation 2	5	0	1	0
Reaction 1C	Operation 3	0	0	1	1000
Pumping 1C	Operation 1	*No time data introduced*			
Reaction 2C	Operation 1	0	0	1	1000
Reaction 2C	Operation 2	1	0	1	0
Reaction 2C	Operation 3	0	0.0003	1	0
Reaction 2C	Operation 4	2	0	1	0
Reaction 2C	Operation 5	0	0.002	1	0

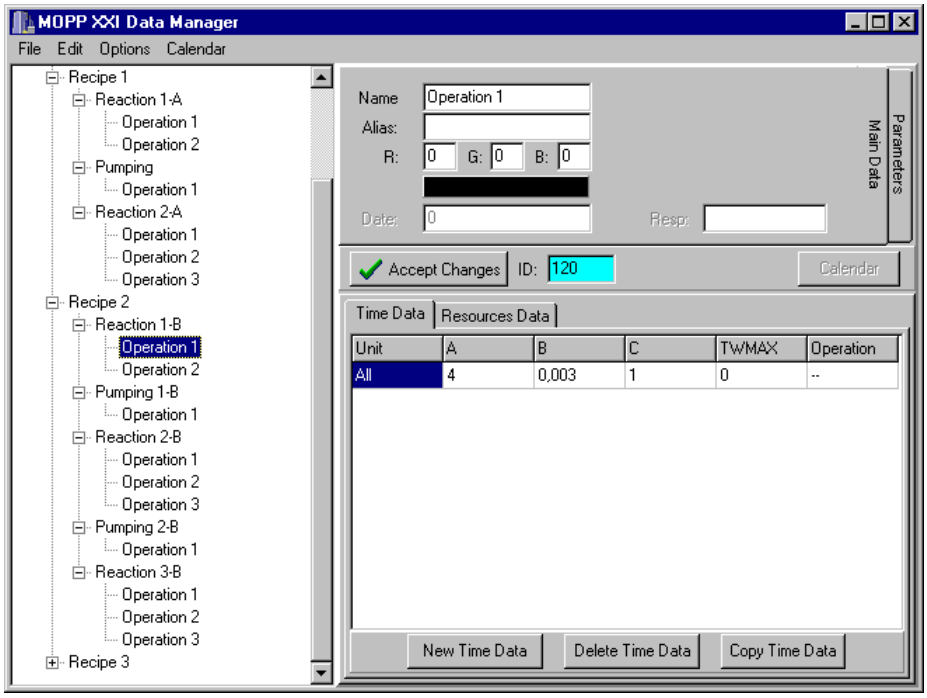
In addition to the time data, resource consumptions for some operations should also be defined. In this case, the electricity resource is required in some operations as it is shown in the following table:

Resource consumption (Electricity)						
Recipe 1						
Stage	**Operation**	**A**	**B**	**C**	**Start**	**End**
Reaction 1A	Operation 1	3	0	1	0	200
Pumping 1A	Operation 1	1	0	1	0	200
Reaction 2A	Operation 2	2	0	1	0	200
Reaction 2A	Operation 3	1	0	1	0	200
Recipe 2						
Stage	**Operation**	**A**	**B**	**C**	**Start**	**End**
Reaction 1A	Operation 1	4	0	1	0	200
Pumping 1B	Operation 1	1	0	1	0	200
Reaction 2B	Operation 2	2	0	1	0	200
Pumping 2B	Operation 1	1	0	1	0	200
Reaction 3B	Operation 2	5	0	1	0	200
Reaction 3B	Operation 3	1	0	1	0	200
Recipe 3						
Stage	**Operation**	**A**	**B**	**C**	**Start**	**End**
Reaction 1C	Operation 2	2	0	1	0	200
Pumping 1C	Operation 1	1	0	1	0	200
Reaction 2C	Operation 2	1	0	1	0	200
Reaction 2C	Operation 3	1	0	1	0	200
Reaction 2C	Operation 4	3	0	1	0	200
Reaction 2C	Operation 5	1	0	1	0	200

The consumption is given as a set of parameter similar to those used by defining the operation time. The quantity consumed in one operatios is expressed as (QC is quantity consumed Q is the batch size):

$$QC = A + B \cdot Q^C$$

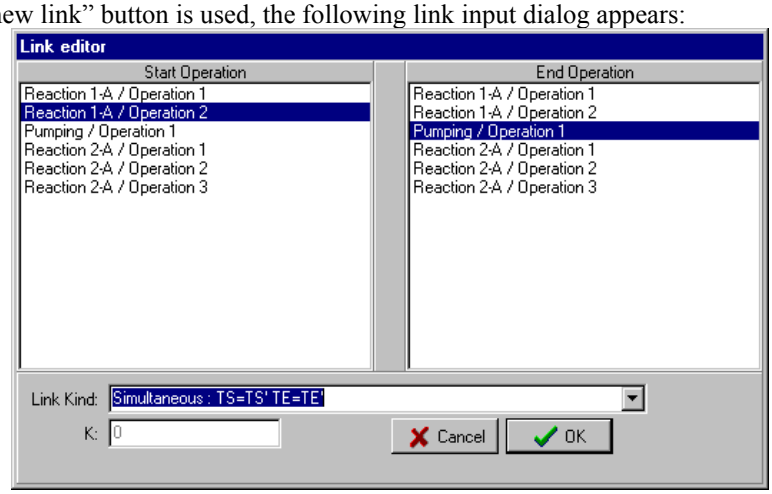
Once introduced all the operations data, the user interface shows the complete trees of the different recipes as it is shown in the following picture:



3.10 Links

The operations introduced don't have any time constraint that relates the different operations of different stages between them. The definition of these interactions is defined at recipe level, in the Operation Links tab.

When the "Add new link" button is used, the following link input dialog appears:



There are four different information fields to be filled:

Start Operation → The first operation to take into account in the constraint

End Operation → The last operation to take into account in the constraint

Link Kind → Constraint to be applied

K →Constant value of the constraint (if applicable)

If TS and TE are the start and end times of the first operation and TS' and TE' are the start and end times of the last operation the different constraints available are:

- Simultaneous: TS=TS' and TE=TE'
- Consecutive: TE+K=TS'
- Start times relationship: TS+K=TS'
- End times relationship: TE+K=TE'
- Consecutive precedence: TE+K <=TS'
- Start Times precedence: TS+K <= TS'
- End times precedence: TE+K<=TE'
- Time inclusion: TS<TS' and TE>TE'

An incorrect link is the most frequent reason why a recipe can be infeasible, for example if two operations are linked by a simultaneous link and the operations times are different, then the recipe is infeasible as you have these constraints:

$$TE - TS = TOP$$
$$TE' - TS' = TOP'$$
$$TS = TS'$$
$$TE = TE'$$

According these constraints TOP should be equal to TOP'. One way to assure that with a simultaneous link this constraint is always valid is to define the TOP only for one of the operations linked and in the other one define a $TW_{max.}$ greater enough. . In this case the constraints are defined as:

$$TE - TS = TOP$$
$$TE' - TS' \leq TW_{\max}$$
$$TS = TS'$$
$$TE = TE'$$

So if TWmax if greater than TOP the constraint will become feasible. This approach is the one that has been used by this test case.

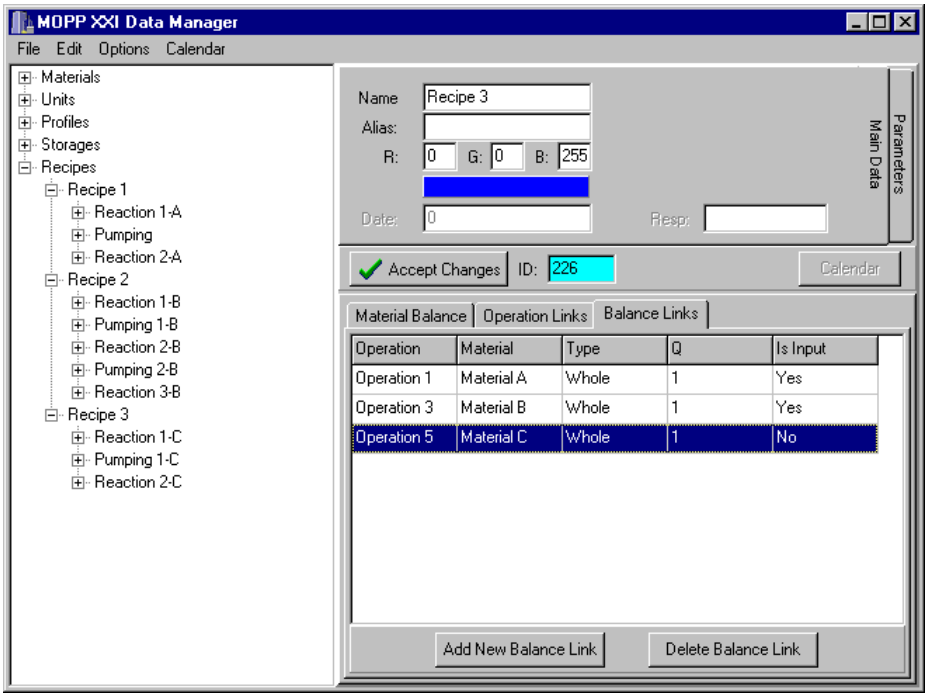
This test case only have SIMULTANEUS links. The detail of the links is listed in the following table:

Links			
Recipe 1			
Start Stage	**Start Operation**	**End Stage**	**End Operation**
Reaction 1A	Operation 2	Pumping 1A	Operation 1
Pumping 1A	Operation 1	Reaction 2A	Operation 1
Recipe 2			
Start Stage	**Start Operation**	**End Stage**	**End Operation**
Reaction 1B	Operation 2	Pumping 1B	Operation 1
Pumping 1B	Operation 1	Reaction 2B	Operation 1
Reaction 2B	Operation 3	Pumping 2B	Operation 1
Pumping 2B	Operation 1	Reaction 3B	Operation 1
Recipe 3			
Start Stage	**Start Operation**	**End Stage**	**End Operation**
Reaction 1C	Operation 3	Pumping 1C	Operation 1
Pumping 1C	Operation 1	Reaction 2C	Operation 1

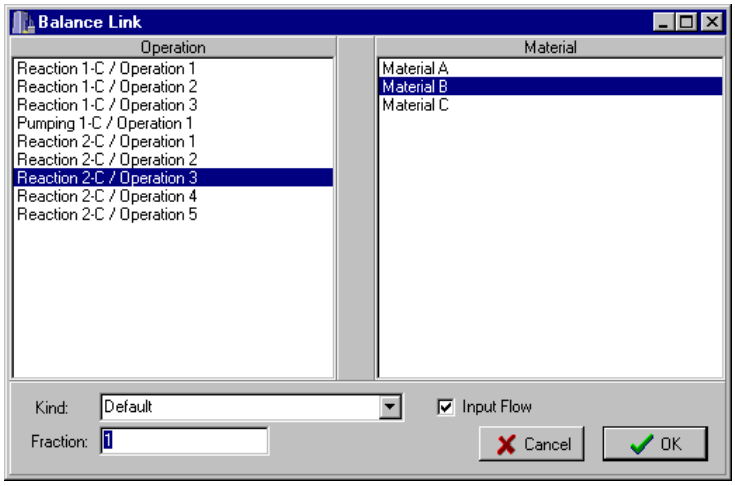
3.11 Balance Links

Up to this moment the global balance for the recipe has been defined. Otherwise there is no exact information about when the inputs and outputs are performed. This information is required in order to calculate the effect in the storage profiles and the application of the storage constraints.

This information is defined in the Balance Links Tab as it is show in the following picture:

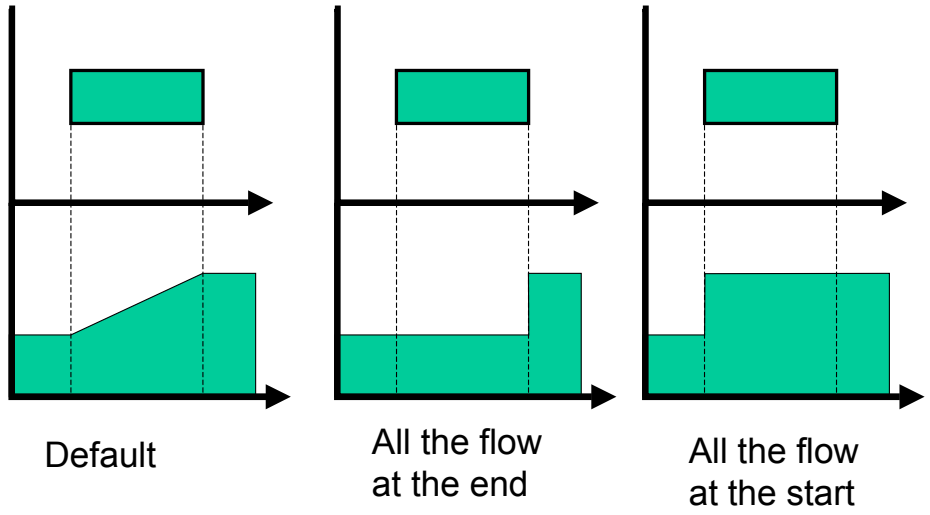


When clicking the Add Balance Link or a current link in the list, the following configuration panel appears:



In the right panel appear all the operations defined in the recipe, in the left one the materials present in the global balance. The idea is to define that a fraction of the quantity related to the global material balance that is a input/output flow in a given operation. In the configuration shown in the previous picture, all the Material B (fraction=1) enters to the process (input flow) during the time elapsed (Kind Default) for all the operation 3 of stage Reaction 2C.

There are three kinds of balance links, they vary in the effect that the input/output of material have in the corresponding profile. The following picture illustrates the difference:



In the tutorial case, the balance links to be defined are all of Default Kind and with fraction 1. The detail of the other parameters of the balance links is included in the following table:

Balance Links			
Recipe 1			
Stage	**Operation**	**Material**	**Is Input**
Reaction 2A	Operation 3	Material A	No
Recipe 2			
Stage	**Operation**	**Material**	**Is Input**
Reaction 3B	Operation 3	Material B	No
Recipe 3			
Stage	**Operation**	**Material**	**Is Input**
Reaction 1C	Operation 1	Material A	Yes
Reaction 2C	Operation 3	Material B	Yes
Reaction 2C	Operation 5	Material C	No

With this information, all the necessary data to define the plant and its possible production recipes is complete.
Next step is to begin to work with all this data in order to create and interact with schedules.

4 Schedule Management

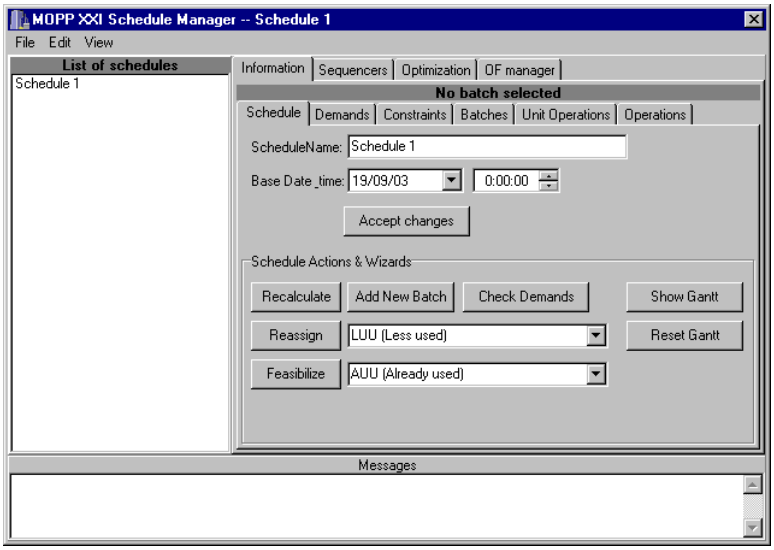
4.1 Introduction

There are two main ways to interact and create an schedule: The schedule manager (plugin planmanager.dll) and the interactive gantt chart (plugins newmoppgantt.dll and plugschactions.dll) . The schedule manager allow the creation of a new empty schedule and its interaction with the sequencer, optimizer and objective function calculation plugins. It also creates and sends data to the gantt-chart objects depending on the gantt-chart plugin.
In other way, the interactive gantt-chart allows interaction with an schedule, changing the constraints, sequence, and assignment of a given schedule.

4.2 The schedule manager

If the schedule manager plugin is connected to the systam, the schedule manager can be activated from the main menu, using the option "*Calculate → Schedule*". The schedule manager window offers this aspect:

The first action that should be performed is the creation of a new schedule. This action is performed by using the menu option "*Edit → New Schedule*". The aspect of the resulting windows is the following one:



When a schedule is selected, there are four main tabs available:

- Information Tab
- Sequencers Tab
- Optimization Tab
- OF Manager Tab

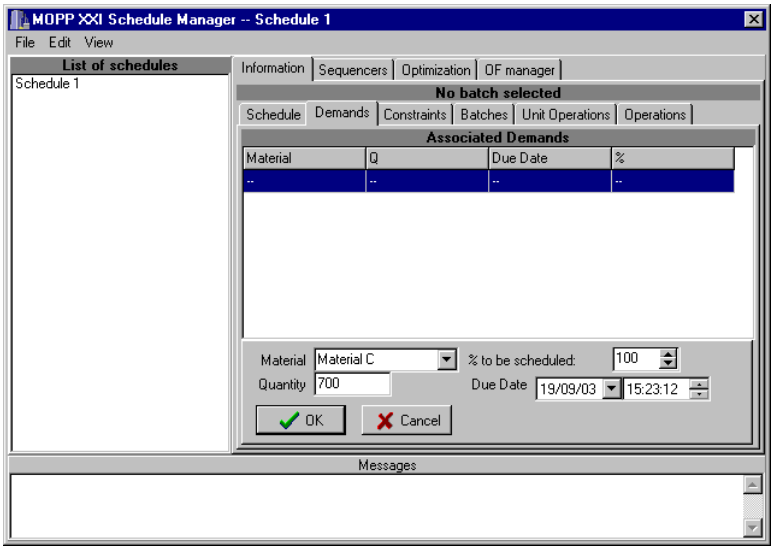
The information tab contains the general information about the constraints applied to the schedule, the Demands assigned, and different manipulation options over the schedule, like racalculation, feasibilization and interaction with the different constraints applied to the schedule at any level.
The Sequencers tab provides interaction with the schedule selected and the sequencer plugins connected to MOPP XXI. It allows configuring and running the different sequencers available in the system.
The optimizers tab provides interaction with the schedule selected and the optimization plugins connected to MOPP. It allows the configuration of the different optimizers and the evaluation of the results obtained after the optimization.
The last tab, the OF manager , allows the interaction with the different OF calculation plugins present in the system and the actual schedule. It also allows the selection of one, or multiple objective functions that can be used in an optimization plugin.

4.3 Generating a schedule

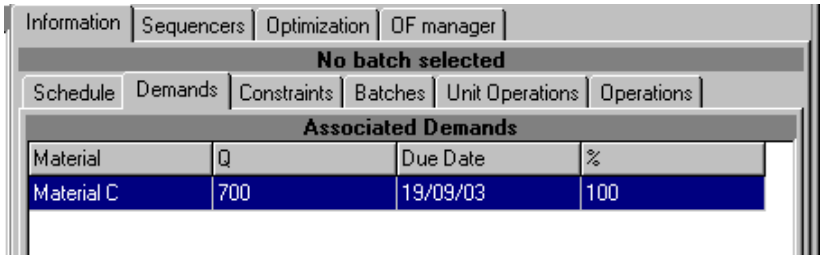
Usually there are two necessary steps to create a schedule:

- Specify the demands
- Run a sequencer

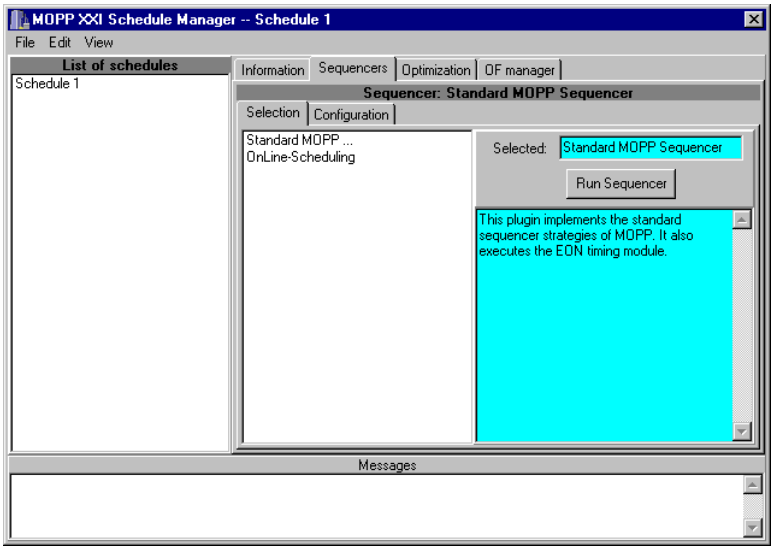
A schedule can also be created manually, using the "*Add New Batch*" button of the information section or simply showing the Gantt and using the Gantt interface to manually add the different batches.
In our case we will generate an schedule authomatically. Thew first Step is specify the demands. To do this go to the Demands tab of the information section and use the "Add new demand" button to specify a demand of 700 units of product C as it is shown in the following picture:



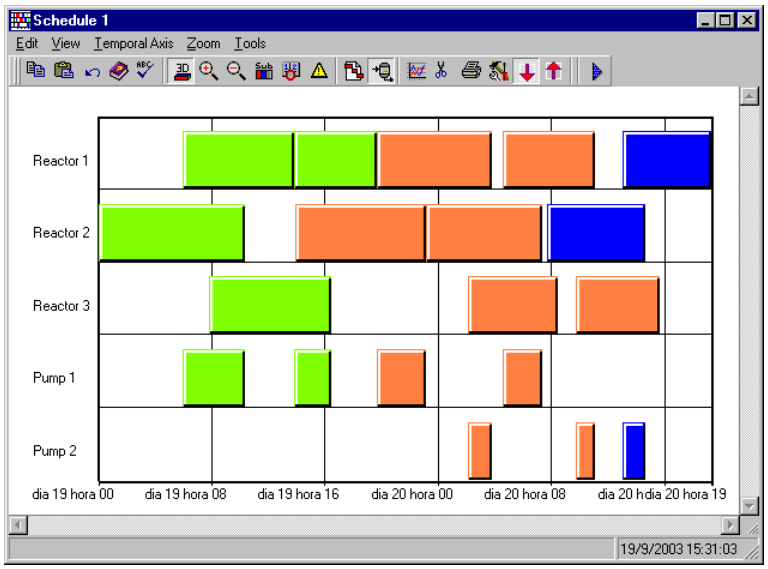
When the demand is accepted (Ok button) the demand will be incorporateds in demand list as it is shown below.



Next step is to generate the schedule according to this demand. This action is performed going to sequencers section. And select the Standard MOPP Sequencer as it is show in the following picture.



In this first case the default parameters of the sequencer will be used, so use the "*Run Sequencer*" button to activate it and generate the schedule. After some calculations, the following gantt-chart should appear:

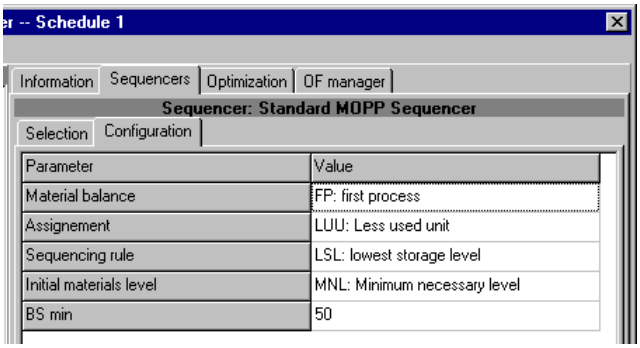


This is a Gantt chart of the schedule created that accomplish with the demand required. When this happens a warning message will appear in the messages window of the schedule manager.. However, the standard sequencer of MOPP is only designed as a useful tool for first approaches of an industrial problem or for its

use in academic cases. Sometimes the schedule generated does not meet all the demands, as the greedy rules used in the standard MOPP sequencer cannot anticipate the future. Nevertheless, the plugin structure of MOPP allows the integration of specific industrial-knowledge-based sequencers that takes into account specific plant politics, and additional constraints and rules, like cleaning an maintenance rules, etc. The standard sequencer of MOPP should only be used in the preliminary study of industrial cases or in academic cases.

4.4 The standard MOPP Sequencer

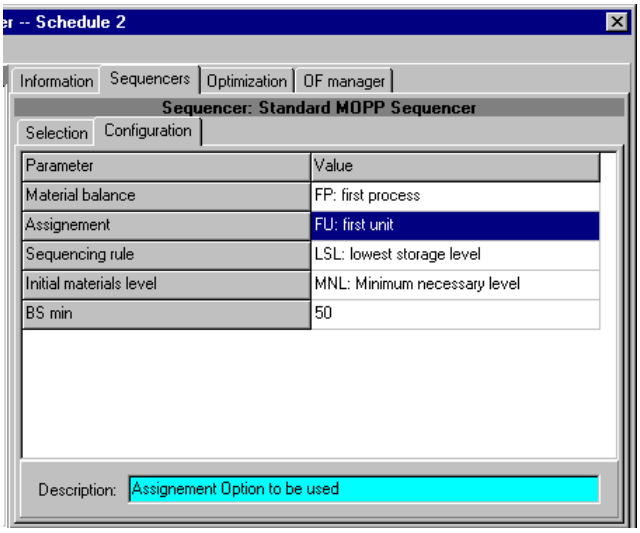
The standard MOPP sequencer have a set of parameters that can be seen selecting the "*Configuration*" tab of the "*Sequencers*" section as shown in the following picture:



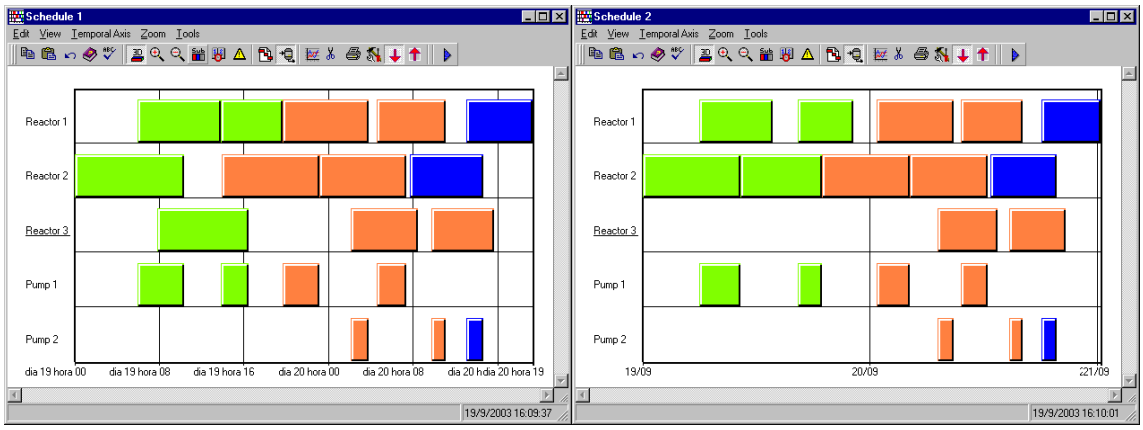
The material balance, Assignment and Sequencing is performed by means of priority rules. The last two parameters allows to control the batch sizing of the resulting batches. The algorithms used in the sequencer are explained in more detail in Cantón (2003). In this tutorial some examples will been shown to illustrate the effects of changing the parameters.

Create a new Schedule (*Edit→New Schedule*) and assign the same demands than in the previous case (Main Tab *Information,* Secondary Tab *Demands,* Button "*Add Existing Demand*").

Then Go to the Sequencers Section And change the Assignement parameter to "*FU: First Unit*" as shown below:

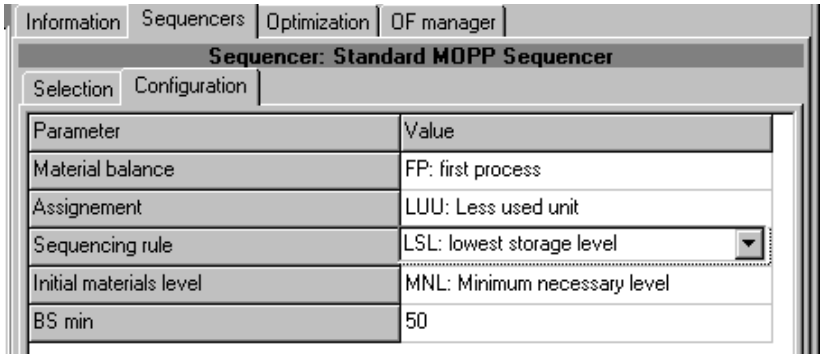


Another schedule will be generated and shown that differs from the previous schedule as it is shown in the following picture:

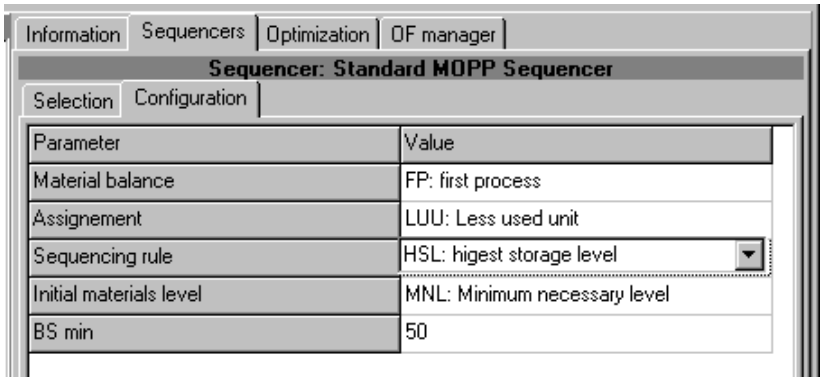


It can be observed that the sequence is maintained, but the assignement of the second batch of Recipe 1 (green ones) have been changed. This is due to the assignment rule applied, in schedule 1 the rule applied was the default one (LUU: Less used unit). In this rule if there are alternative units as in the case of the first task of the Recipe1. The unit used is the one that have been used less times before, so if the first batch uses Reactor2, the second batch uses Reactor 3. On Schedule 2, the rule used is (*FU: First unit)* The first alternative unit defined in the recipe is used, so all the batches of Recipe1 uses Reactor 2 for the first Task.

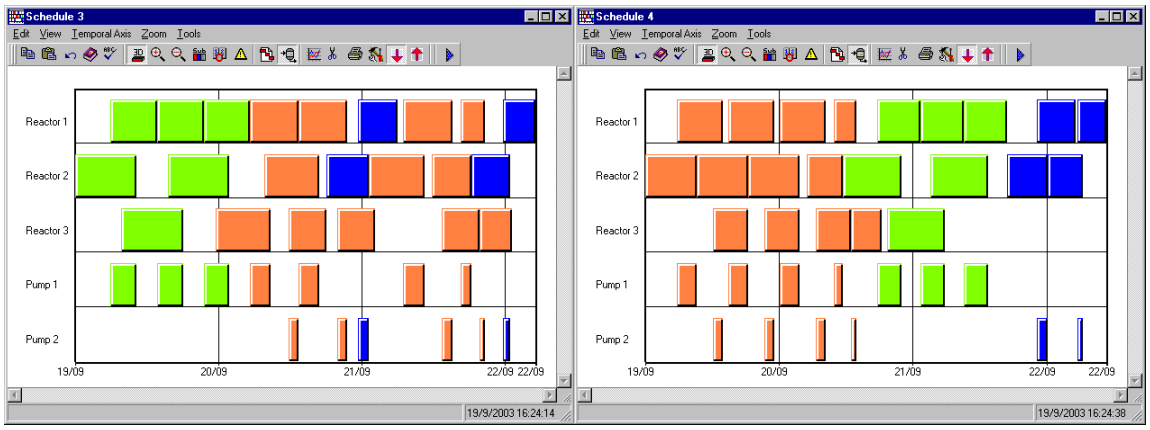
Let's See another example, Create a new schedule (Schedule 3) with a demant of 1300 units of product C and run the sequencer with the following configuration:



Create also another schedule (Schedule 4) with the same demand of 1300 units of Product C and the following configuration:



The resulting Gantt-Charts will be the following ones:



It can be observed that when using the LSL rule (Schedule 3), the batches of Recipe 3 are produced as soon as possible, when enough intermediate is available. In the other hand, on Schedule 4 where HSL rule is used, the batches of Recipe 3 are produced at the end, when all the intermediate material has been generated. The Sequencing rule basically affects at the final sequence of batches, but also affects at the batch sizes as can be checked in this example.

For the following, example create two other schedules (Schedule 5 and Schedule 6) With a demand of 300 units of **Material A**. The configuration parameters for Schedule 5 are:

Information | Sequencers | Optimization | OF manager

Sequencer: Standard MOPP Sequencer

Selection | Configuration

Parameter	Value
Material balance	FP: first process
Assignement	LUU: Less used unit
Sequencing rule	LSL: lowest storage level
Initial materials level	MNL: Minimum necessary level
BS min	50

And for Schedule 6

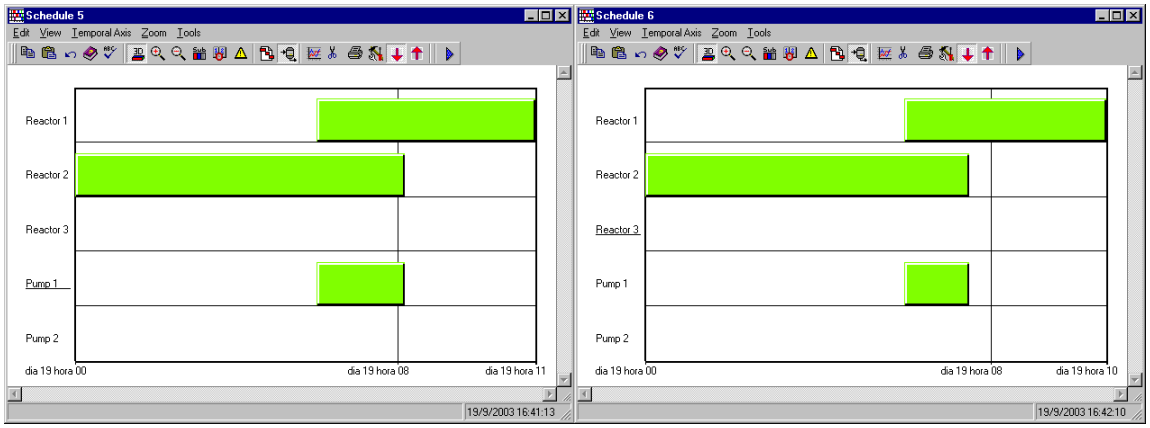
Information | Sequencers | Optimization | OF manager

Sequencer: Standard MOPP Sequencer

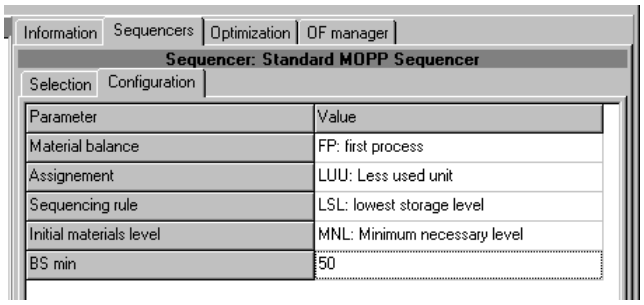
Selection | Configuration

Parameter	Value
Material balance	FP: first process
Assignement	LUU: Less used unit
Sequencing rule	LSL: lowest storage level
Initial materials level	MNL: Minimum necessary level
BS min	10

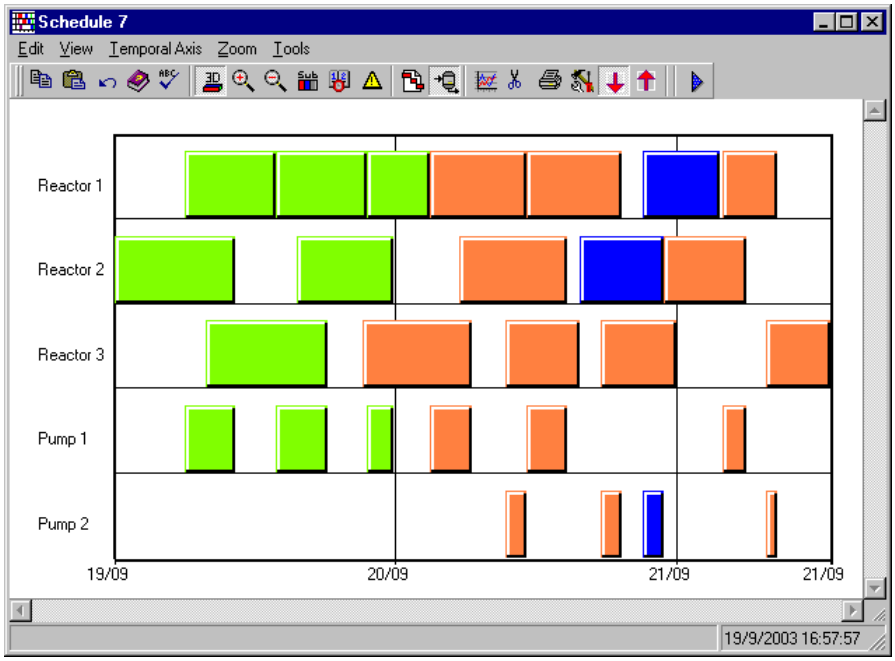
The resulting schedules are:



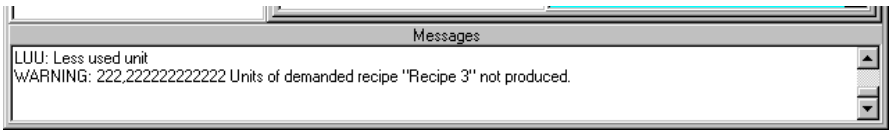
The important variable here is the batch Size, In Schedule 5 the batch size of the batch created is of 437,5 units, whilst in Schedule 6 the batch size is 300. In schedule 5, the batch size exceeds the demand because according to the parameter BS min the minimum acceptable batch size for a batch is the 50% of the nominal batch size. As the nominal batch size for Recipe 1 with this unit assignment is calculated as 875 units, the minimum acceptable batch size is of 437,5 units, which is the batch size used. On the Schedule 6 this minimum is only 10% of nominal and a batch of only 300 units can be created.
This information can be examined by selecting the batch on the Gantt-chart (Shift-Mouse click on the batch) and asking the information of the selected batch (Alt+Enter) as it will be seen on the section about Gantt-chart manipulation options.
As it has been explained before, it is possible that the schedule generated doesn't meet all the demands due to the limitations of the greedy algorithms used. This can be observed in the following example. Create a new Schedule (Schedule 7) with a demand of 1000 units of Material C and with the following sequencer configuration.



The resulting Schedule is the following one:



When using the standard Sequencer is very important to take care about the messages that appear in the message window:



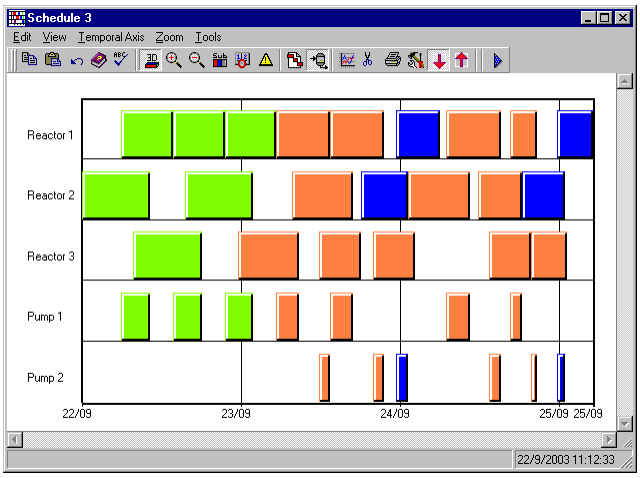
In this case, the sequencer shows that it has not produced 222,22 units of Recipe 3. Why?, well the reason is the combination of rules/batch sizes used. In one hand the algorithm knows that it needs to produce a batch of at least 222,22 units to meet the demand of Material C, in the other hand it only have enough raw material to produce 222,22 units. There is another constraint involved: as a parameter is set that the minimum batch size is 50% of the nominal batch size, as the nominal batch size of Recipe 3 is 833,33 units then the minimum batch size is 416,65. This batch size cannot be produced because there is not enough intermediate material to produce it.
Anyway, the system warns the user, and the user can take the appropriate decision for producing the remaining 222 units of material C.

4.5 Gantt chart interaction

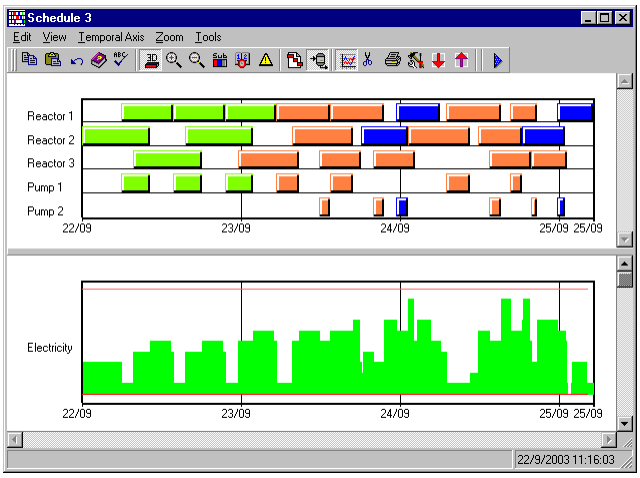
MOPP XXI is provided with and EGC (Electronic gantt-chart) as the main tool for interaction and manipulation of an schedule. The Gantt chart allows the manipulation of the sequence, the assignment and the constraints associated to the schedule, as well as obtaining information about the schedule.

4.5.1 View definition

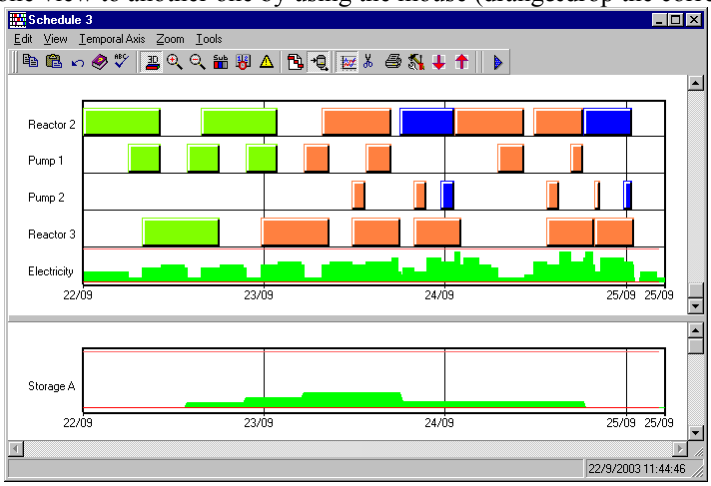
As an example of gantt interaction let's start with the results obtained for schedule 3 of the previous section:



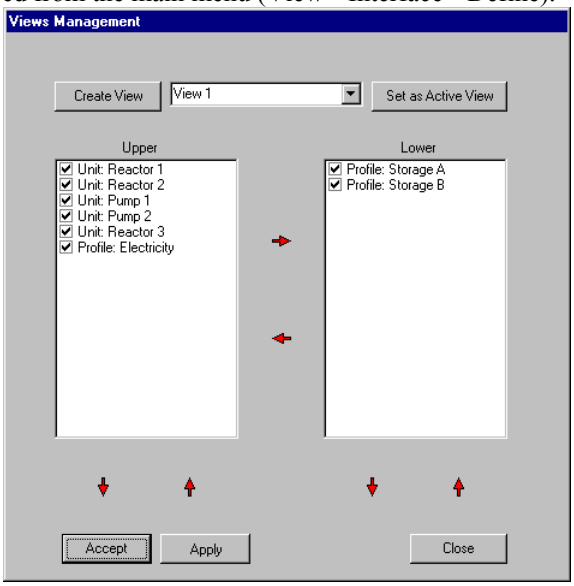
The first thing to do is to show the information about the profiles. This is done by pressing the icon  (See profiles) and icon  (Only Top view) as a result we now see the electricity consumption profile:



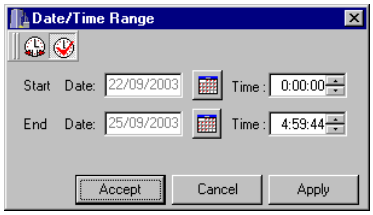
Using the scroll bars it can be show the different profiles available (Electricity, Storage A and Storage B). The EGC is divided intwo views (Top and bottom view) which can be activated/deactivated by using the buttons  . Each view can contain a mix of profiles and units. As a default configuration the units are shown in the top view and the profiles are shown in the bottom view. The user can drag & drop the different units from one view to another one by using the mouse (drang&drop the corresponding label).



Another way to configure what will be appear in the top/bottom view is by using the view configuration window, that can be accessed from the main menu (View->Interface->Define).



Additionally, at any moment is possible to activate/deactivate the unit and profile rows by using the buttons  . These buttons allows to show/hide the units and the profiles of all the EGC.
The horitzontal scale can also be set up. By using the menu option Temporal Axis->Define it will appear the following configuration window:



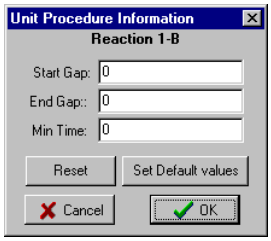
That it is used to define the visible range of time. This aspect can also be manipulates by using the zoom options (Buttons ).

Finally the buttons  and  also allows the manipulation of the EGC aspect, by activating/deactivating the labels associated to the operations, the 3D effect of the EGC “boxes” and the switch between see the operation detail or not.

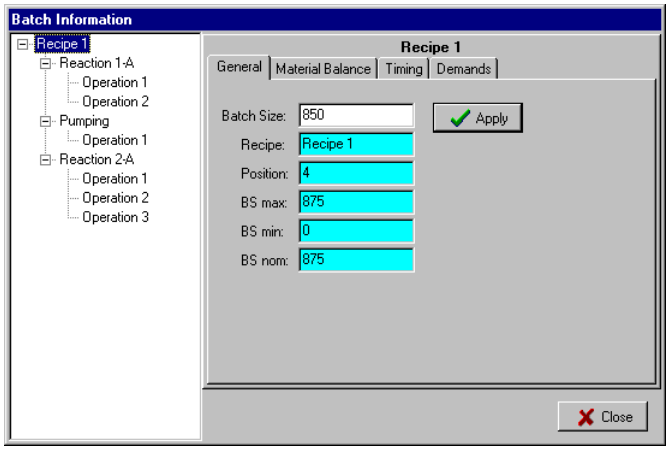
4.5.2 Obtaining Information

The Gantt chart allows obtaining the information at different levels: Task, Batch, Unit and Profile. The ECG has a context sensitive menu associated to the mouse right click. Depending on the object selected, it will appear more or less options in that menu.

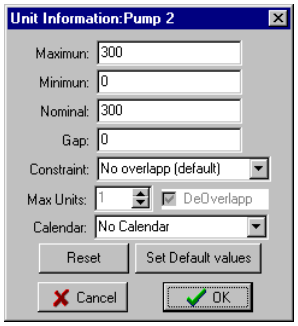
To select a unit procedure, simply left-click the box representing it. In order to select a whole batch, hold the shift key while left-click on a unit procedure pertaining to the batch. If you want to select a unit or a profile, push the mouse left button over a unit/profile line, not used by any of the “boxes” that represents the stages, the label corresponding to the unit/profile will appear underlined. Once something is selected, the properties of that selection can be accessed by the properties menu option or by clicking Alt+Enter. Depending on what is a selected, different window will appear. If a task is selected it will appear a window allowing the manipulation of the constraints associated to the stage:



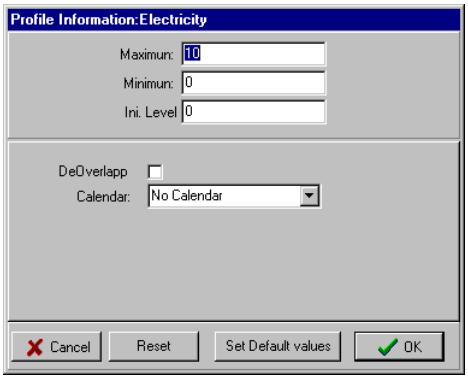
If a batch is selected, it will appear a window that allows the manipulation of the batch constraints, and different levels:



When a unit is selected, then appears the unit information window, which allows the manipulation of the constraints associated to the unit.

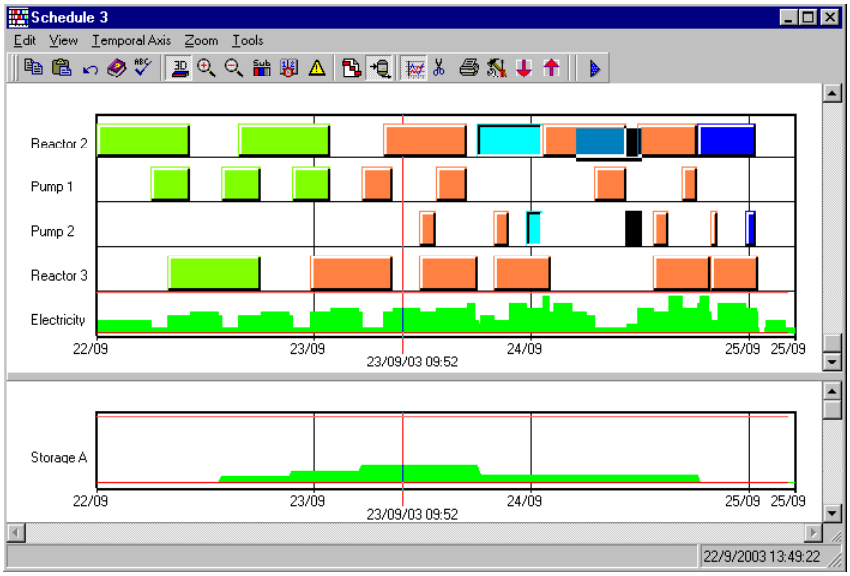


Finally, if a profile is selected, it will appear the profile information window, which allows the manipulation of the profile constraints.



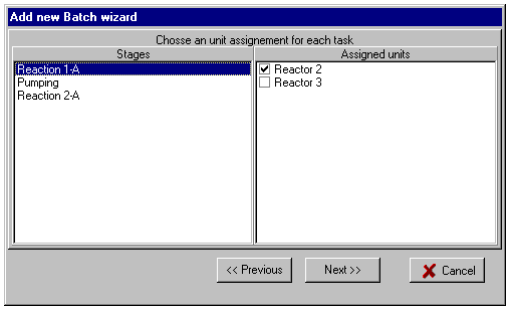
4.5.3 Basic manipulation

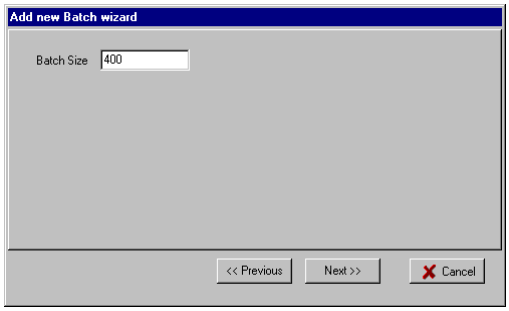
The basic changes that can be made over a schedule is to change the sequence and the assignment. Drag & drop of the stages can easily perform these actions. To start the drag maintain the left mouse button pressed over a task, then drag the task to the desired time/unit whilst the left button of the mouse is pressed and leave the button when you reach the desired position.



A selected batch can also be copied and pasted whit the same functionality of a word processor. To copy a batch, select it (shift+left mouse click) and use Ctrl+C, to paste it Ctrl+V and to delete it Del.

New batches can also be created easily using create new batch button ( ). This functionality shows a wizard that allows the creation of a new batch of any recipe.



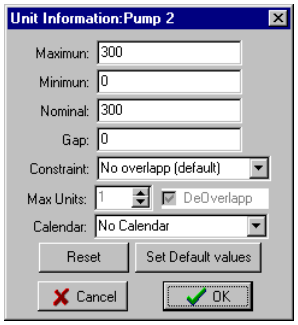


This functionality is especially useful for creating a whole schedule from an empty Gantt chart.

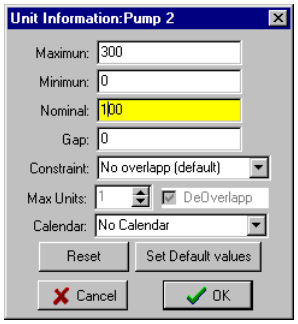
4.5.4 Constraint manipulation

One of the most powerful features of MOPP XXI EGC is the easy constraint manipulation that it offers. Each time a schedule is created, a copy of the constraints depending on unit, profile and storage are also generated and assigned to the schedule. This fact allows the creation of *what if* scenarios of the same schedule under different constraints.
Let's start with the previous schedule :

If The constraints associated to Pump2 are show we foud this information:

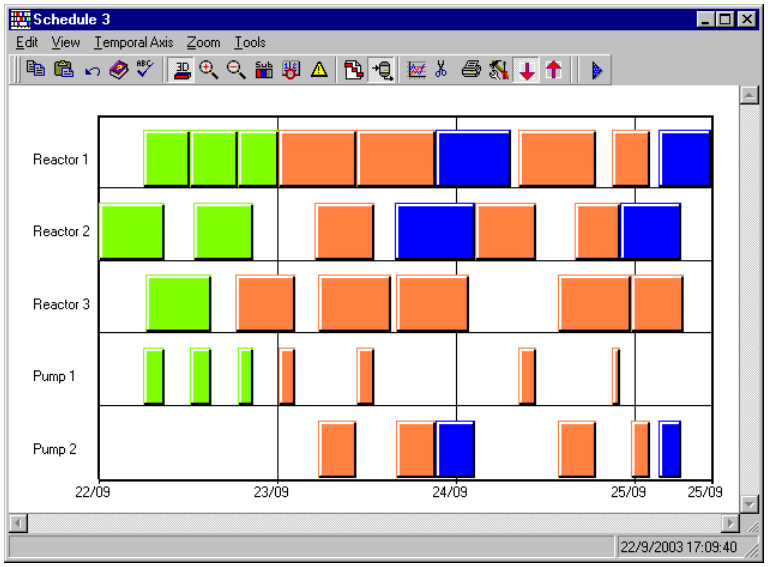


Let's imagine that there is a problem in the pump that decreases its nominal capacity to only 100, ¿what will happens to the schedule? Well, to know that first modify the nominal for the pump to 100:

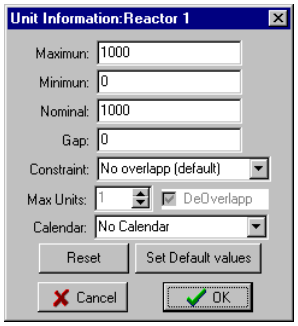


Then press the OK button and ask for recalculation. (Button )

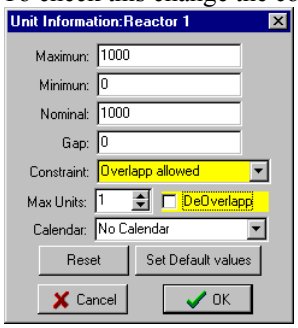
The result is the following EGC



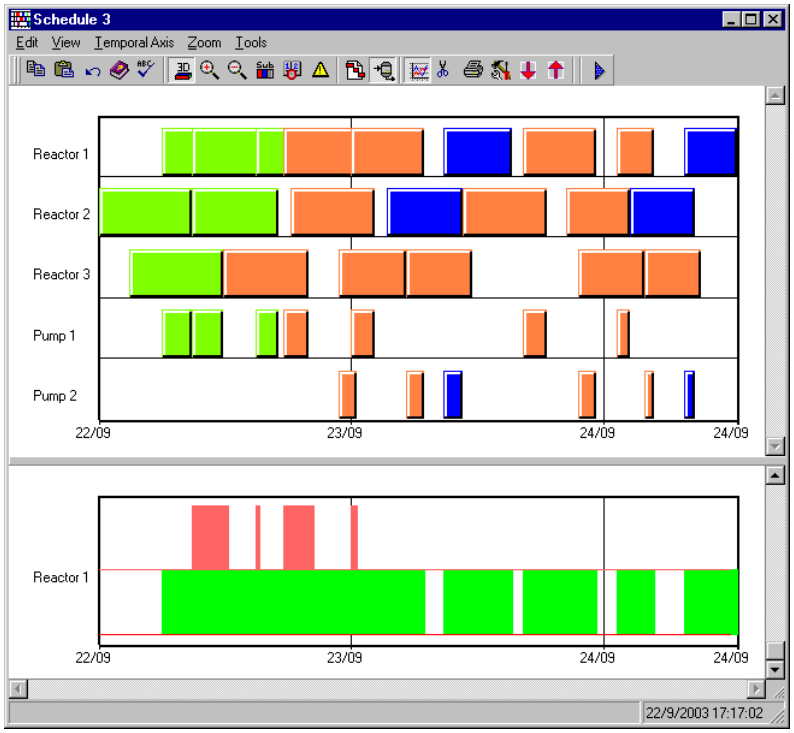
The time needed for the operations carried out in Pump2 are now much longer, as the operation time depends on the nominal capacity of the pump. Now change again the nominal capacity of Pump2 to 300 and open the properties for Reactor 1



Reactor 1 seems a bottleneck in the Gantt chart, we want to examine what will happen if there is more that one reactor available for that operation. To check this change the constraints to the following values:

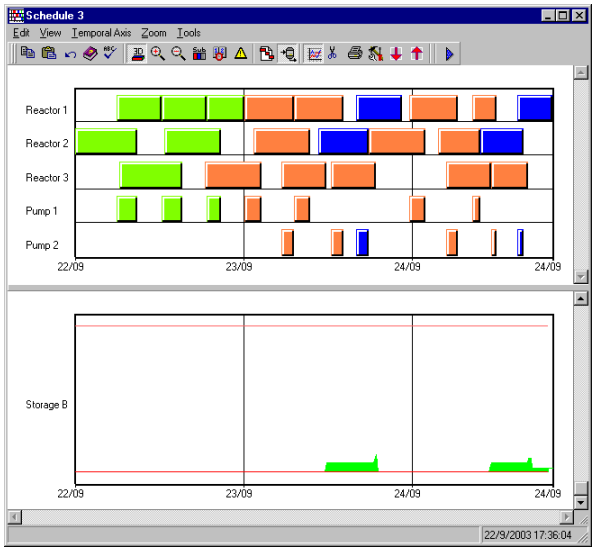


After recalculation, the result is the following one:

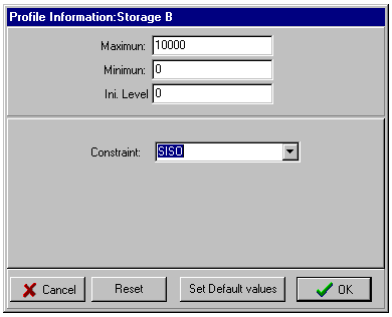


Some boxes are overlapped, and a new profile, called Reactor 1 has appeared showing the magnitude of the overlapping. As it can be observed, there are little overlapping in the Reactor 1 and the makespan have been decreased a little.
Now, lets set the constraints of Reactor 1 again to its first values (Button *Set Default Values* of the information window).

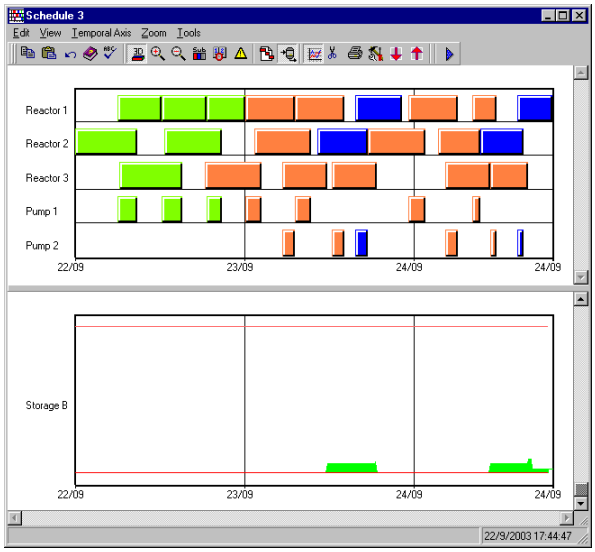
Look at the profile of Storage B



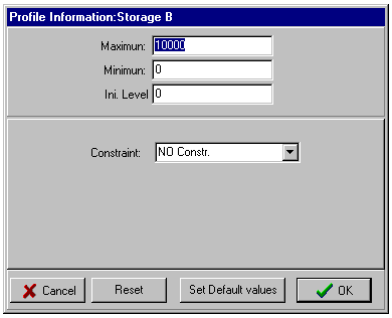
There are some small peaks due to the default constraint defined in Storage B which is SIO, this means, that Storage B doesn't accept a charge and a discharge simultaneosly. This is also the reason for some of the gaps present in Reactor 1 and Reactor 2. To see the efect of this constraint change the constraint of Storage B from SIO to SISO



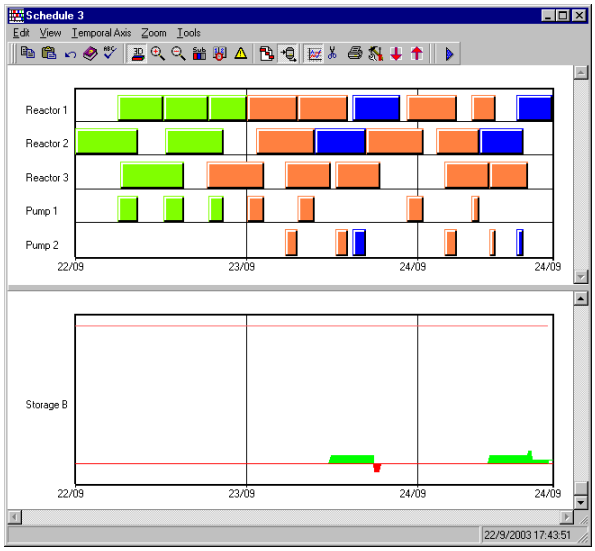
The result after asking for recalculation is



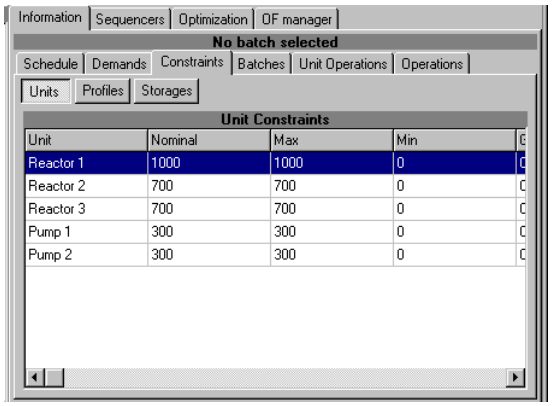
Note that there are still some gaps in the reactors, let's see what happens if the Storage constraint is changed to No constraint



The result is



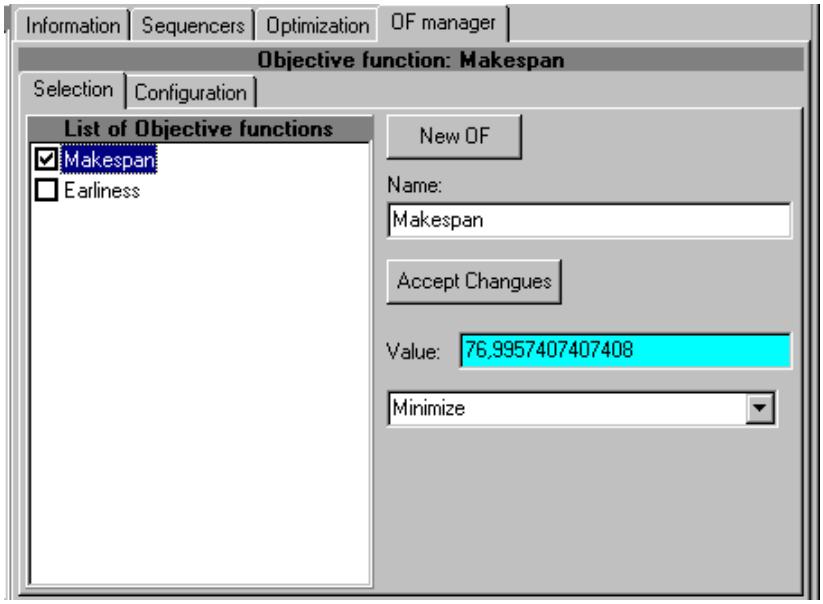
Deleting the constraint makes the storage profile go below zero which is infeasible.
The other possible constraints (associated to the stage and to the batch) works in the same way.
Feel free to experiment. There are more details of these kinds of constraints in the model reference.
All these functionalities and constraint manipulation can also be done in the information tab of the schedule manager.



From there you can also create/copy/delete batches, recalculate the schedule and manipulate the constraints associated to any level of the schedule.

4.6 Objective function configuration

Going back to the schedule manager, there is the OF manager tab.



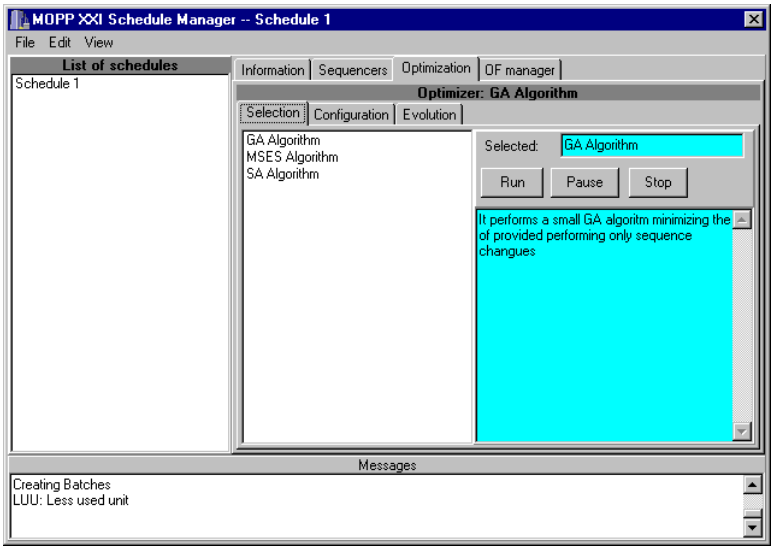
This tab allows the selection of one or more of the present objective functions for maximization or minimization. The objective functions can also have specific parameters that can be configured in the Configuration Tab.
New objective functions can be easily added to the system by programming a new plugin that implement the specific calculation required.
All the standard MOPP optimizers work only with one objective function. Nevertheless the system is prepared to work with optimizers that require more than one objective function.

4.7 Optimization algorithms

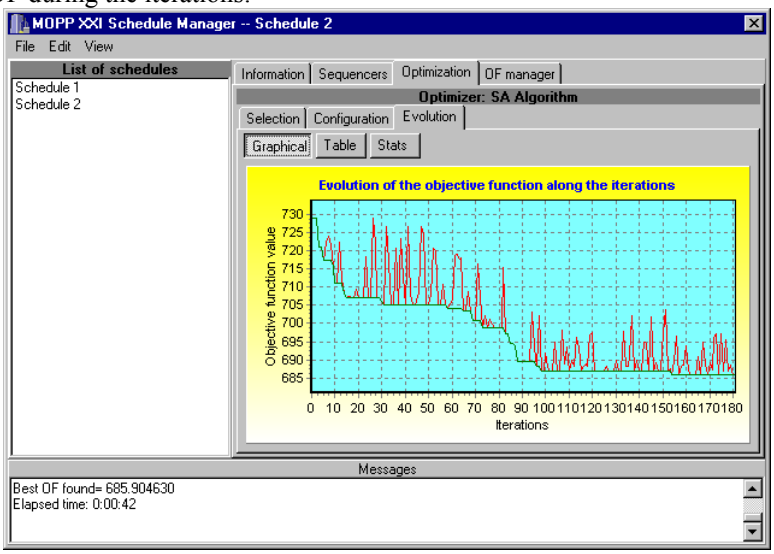
The last aspect to see in this brief tutorial is realted with the manipulation of the optimization algorithms, as all the other aspects and functionalities of MOPP XXI, the optimization algorithms are implemented as plugins. There are three standard optimization algorithms available:

Moppsa.dll	SA optimization (Simulated annealing)
Moppga.dll	GA optimization (Genetic algorithms)
Moppmses.dss	MSES optimization (Mixed stochastic enumerative search)

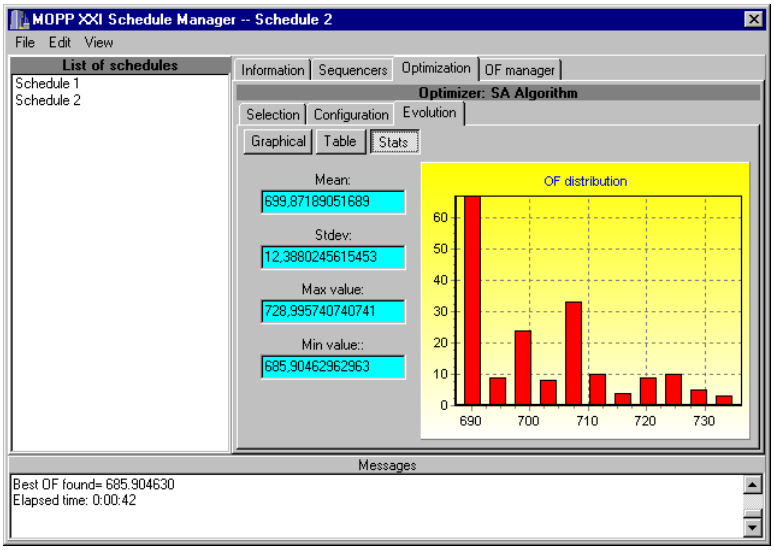
Each of them use only one objective function, and are capable of sequence and assignment changes in a given schedule (Number of batches and their batch sizes are maintained).
All the optimization algorithms works in the same way, the optimization works directly on the sequence of batches, and the assignment is generated after the sequence by means of a set of assignment rules similar to those used in the standard MOPP sequencer.
All the optimizers are accessed through the Optimization tab of the schedule manager:



There are three tabs in the Optimization section, the first one allows to chose the active optimizer to be used and run/stop it. The second one allows the configuration of the specific parameters of the selected optimization algorithm, and the last one allows the exploration of the results.
The procedure to use an optimization algorith is always the same:
Select the schedule to optimize
Select the objective function for maximization or minimization
Select the optimization algorithm to be used
Configure the parameters of the optimization algorithm
Run the optimization algorithm
Check the results
The results are given in the message box (the best OF found) and in the Evolution Tab, that shows the evolution of the OF during the iterations:



and also the distribution of this objective function in the search space used:



Each case works better with a different optimization algorithm, additionally, specific optimization algorithms can be added to the system by programming a new optimization plugin.

5 Conclusions

This tutorial has only shown the basic functionality of MOPP XXI. From this point feel free to explore all the options available offered by the different plugins. If you want to go deeper in the algorithms and background of MOPP XXI feel free to download and read the Ph.D. Thesis: *Integrated support system for planning and scheduling of batch chemical plants* That can be downloaded from http://www.tdx.cesca.es/TDX-0707103-125200/ .