
AIMMS Tutorial for Professionals - Production and Transport Overviews

This file contains only one chapter of the book. For a free download of the complete book in pdf format, please visit www.aimms.com

Copyright © 1993–2011 by Paragon Decision Technology B.V. All rights reserved.

Paragon Decision Technology B.V.	Paragon Decision Technology Inc.	Paragon Decision Technology Pte.
Schipholweg 1	500 108th Avenue NE	Ltd.
2034 LS Haarlem	Ste. # 1085	80 Raffles Place
The Netherlands	Bellevue, WA 98004	UOB Plaza 1, Level 36-01
Tel.: +31 23 5511512	USA	Singapore 048624
Fax: +31 23 5511517	Tel.: +1 425 458 4024	Tel.: +65 9640 4182
	Fax: +1 425 458 4025	

Email: info@aimms.com
WWW: www.aimms.com

AIMMS is a registered trademark of Paragon Decision Technology B.V. IBM ILOG CPLEX and sc CPLEX is a registered trademark of IBM Corporation. GUROBI is a registered trademark of Gurobi Optimization, Inc. KNITRO is a registered trademark of Ziena Optimization, Inc. XPRESS-MP is a registered trademark of FICO Fair Isaac Corporation. MOSEK is a registered trademark of Mosek ApS. WINDOWS and EXCEL are registered trademarks of Microsoft Corporation. $\text{T}_{\text{E}}\text{X}$, $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, and $\text{A}_{\text{M}}\text{S}_{\text{L}}\text{A}_{\text{T}}\text{E}_{\text{X}}$ are trademarks of the American Mathematical Society. LUCIDA is a registered trademark of Bigelow & Holmes Inc. ACROBAT is a registered trademark of Adobe Systems Inc. Other brands and their products are trademarks of their respective holders.

Information in this document is subject to change without notice and does not represent a commitment on the part of Paragon Decision Technology B.V. The software described in this document is furnished under a license agreement and may only be used and copied in accordance with the terms of the agreement. The documentation may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior consent, in writing, from Paragon Decision Technology B.V.

Paragon Decision Technology B.V. makes no representation or warranty with respect to the adequacy of this documentation or the programs which it describes for any particular purpose or with respect to its adequacy to produce any particular result. In no event shall Paragon Decision Technology B.V., its employees, its contractors or the authors of this documentation be liable for special, direct, indirect or consequential damages, losses, costs, charges, claims, demands, or claims for lost profits, fees or expenses of any nature or kind.

In addition to the foregoing, users should recognize that all complex software systems and their documentation contain errors and omissions. The authors, Paragon Decision Technology B.V. and its employees, and its contractors shall not be responsible under any circumstances for providing information or corrections to errors and omissions discovered at any time in this book or the software it describes, whether or not they are aware of the errors or omissions. The authors, Paragon Decision Technology B.V. and its employees, and its contractors do not recommend the use of the software described in this book for applications in which errors or omissions could threaten life, injury or significant loss.

This documentation was typeset by Paragon Decision Technology B.V. using $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ and the LUCIDA font family.

Chapter 11

Production and Transport Overviews

In this chapter you will build two end-user pages that display the solution corresponding to a single ‘roll’ in the rolling horizon process. The first page, the *Production Overview* page, concentrates on the optimal production and maintenance schedule for every period in the current planning horizon. The second page, the *Transport Overview* page, provides not only the optimal transport patterns from the factories to the distribution centers, but also the corresponding stock overviews for all locations considered.

This chapter

11.1 Extending the model tree

Whenever you build a professional user interface, it is quite natural to introduce additional identifiers to support such an interface. For instance, an element parameter defined over the predefined set of `AllColors` can be used to change the color of numbers when they drop below a particular threshold value. Another possibility is the introduction of parameters to control the scrolling mechanism of a Gantt chart. Yet another option is an identifier to control whether or not a particular object appears at all depending on data elsewhere in your application.

*Needing
additional
identifiers*

You should now introduce five extra sections in your model tree corresponding to the five end-user overview pages already introduced in the **Page Manager**. All new page-specific identifiers introduced can then be inserted into the appropriate section. The updated tree structure is shown in Figure 11.1.

*Introducing
extra model
sections*

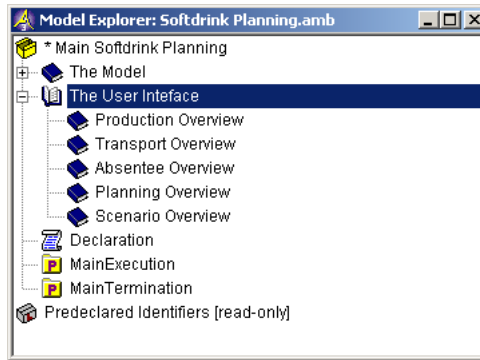


Figure 11.1: Subdividing the The User Interface section

11.2 The Production Overview page

In this section you will construct the entire page as shown in Figure 11.2. Each page object will be treated in a separate subsection.

Viewing the entire page

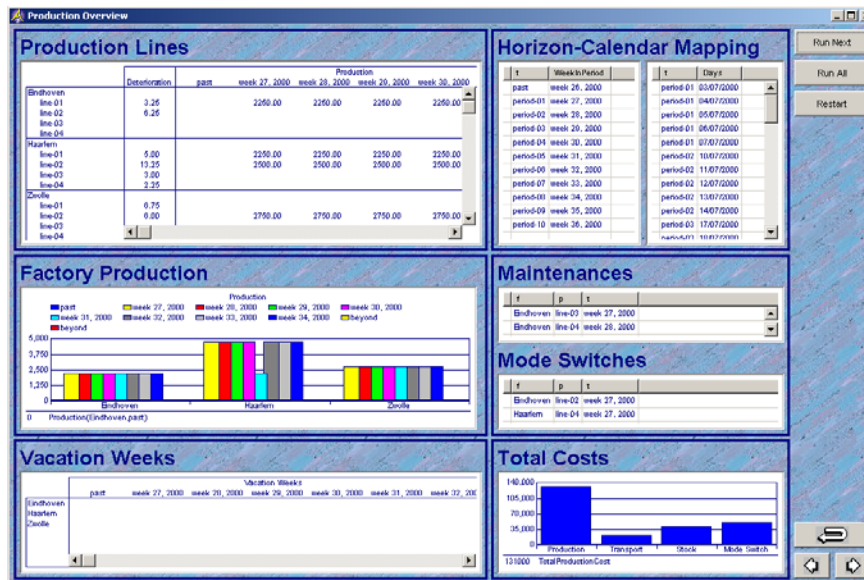




Figure 11.2: The completed *Production Overview* page

11.2.1 Execution buttons

The first execution button you will add is designed to execute a single step in the rolling horizon process. This allows you to track the behavior of the model step by step. To create the **Run Next** button you should perform the following actions:

The Run Next button

- ▶ open the *Production Overview* page in **Edit** mode,
- ▶ press the **New Button** button  on the toolbar,
- ▶ drag and create a small rectangle in the upper right corner of the page,
- ▶ specify "Run Next" (with the quotes) in the 'Title' edit field,
- ▶ press the **Actions** tab,
- ▶ select the 'Run' action,
- ▶ press the **Add** button,
- ▶ select the 'Procedure' option (not the 'Page Procedure' option),
- ▶ use the **Wizard** button  to select the procedure RollHorizonOnce,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

The second execution button to be added is designed to execute the entire rolling horizon process from the current point forward. Just repeat the steps in the previous paragraph while creating the the **Run All** button, but select the procedure RollHorizonToEnd.

The Run All button

The third execution button is the **Restart** button which activates the procedure MovePlanningIntervalToStartOfCalendar. Following the execution of this procedure you can use either of the previous two execution buttons to execute part or all of the rolling horizon process. Instead of creating the button from scratch, as in the previous two paragraphs, you could use the 'copy and paste' facility as described in the following steps:

The Restart button



- ▶ in **Edit** mode, select the **Run All** button by clicking on it,
- ▶ press the **Copy** button  on the toolbar,
- ▶ press the **Paste** button  on the toolbar (the mouse cursor will change as shown in Figure 11.3),
- ▶ use the mouse cursor to position the new button underneath the **Run All** button,
- ▶ click the left-mouse button to confirm the position of the new button,
- ▶ double-click the left-mouse button to open the **Button Properties** dialog box of the new button, and
- ▶ modify the button properties as appropriate.



Figure 11.3: The mouse cursor after having pressed the **Paste** button

11.2.2 The production lines table

In the first table on the *Production Overview* page you will include three identifiers, namely:

Three identifiers in one table

- the actual production level by factory, production line and time period,
- the number of working days in each week, and
- the current deterioration level associated with each production line.


The actual level of production will be equal to potential production whenever a production line is in use. Create a new declaration section *Production Overview Declaration* in the *Production Overview* section, and insert the following parameter declaration:

Actual production level

```
PARAMETER:
  identifier   : ActualProduction
  index domain : (f,p,t)
  unit        : hl
  definition   : PotentialProduction(f,p,t) * ProductionLineInUse(f,p,t)
```


The first part of the table can be created by executing the following steps:

Creating a table

- ▶ ensure that the *Production Overview* page is in **Edit** mode,
- ▶ press the **New Table** button  on the toolbar,
- ▶ drag and create a rectangle that matches the desired table size on your page,
- ▶ in the **Identifier** wizard select the parameter `ActualProduction(f,p,t)`,
- ▶ press the **Next** button, and
- ▶ press the **Finish** button.

To add the identifier `DeteriorationLevel(f,p)` as the first column of this new table you should perform the following actions:

Adding an identifier

- ▶ select the existing table object,
- ▶ press the **Properties** button  on the toolbar,
- ▶ select the **Contents** tab,
- ▶ press the **Add** button,
- ▶ select the identifier `DeteriorationLevel(f,p)` using the **Identifier** wizard,
- ▶ press the **Next** button,

- ▶ uncheck the 'Automatic split row/column' checkbox,
- ▶ select the 'split line' entry that pops up in the listbox (see Figure 11.4),
- ▶ press the **Down** button,
- ▶ press the **Finish** button,
- ▶ press the **Up** button to display the identifier DeteriorationLevel as the *first* column, and
- ▶ press the **OK** button.

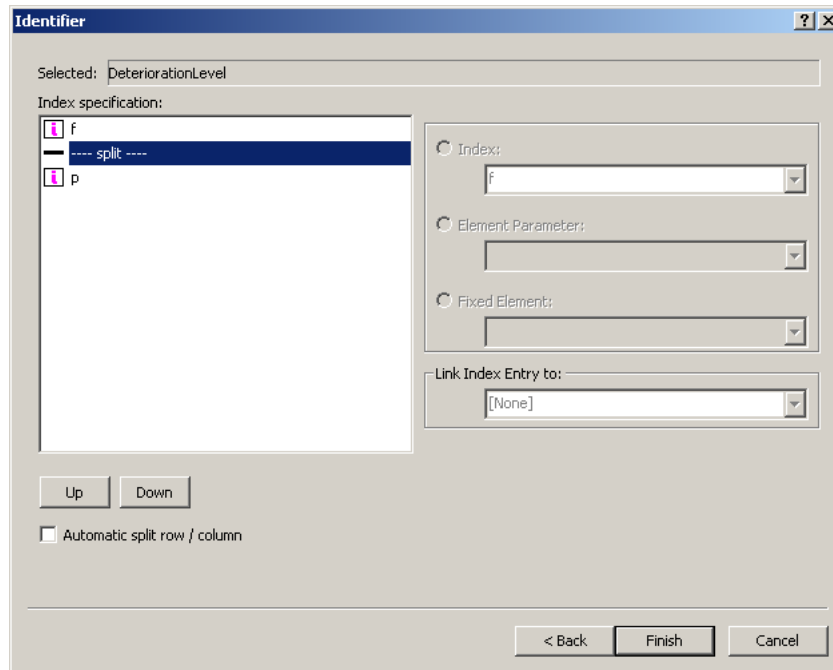


Figure 11.4: Specifying the row and column domain

If you had not moved the split line, AIMMS would have used the index f for rows and the index p for columns. However, by moving the split line, both indices can be used as row indices conforming to Figure 11.5.

Moving the split line

Following the routine specified above, you should now add the identifier ActualNumberOfDaysInPeriod(t) as a new row in the table. The table on your screen should then look like the one shown in Figure 11.5.

Adding another identifier

	DeteriorationLevel	ActualProduction								
		past	period-01	period-02	period-03	period-04	period-05	period-06	period-07	period-08
Eindhoven										
line-01	2.3									
line-02	6.0									
line-03										
line-04										
Haarlem										
line-01	4.0									
line-02	12.3									
line-03	2.8									
line-04	2.0									
Zwolle										
line-01	6.5									
line-02	5.0									
line-03										
line-04										
ActualNumberOfDaysInPeriod										

Figure 11.5: The initial production overview table

The ‘period’ references in the table are somewhat abstract and not meaningful. In AIMMS you can change these references using a string parameter. You should first create this string parameter in the section Production Overview Declarations.

Creating week labels...

```

STRING PARAMETER:
  identifier : PeriodDescription
  index domain : (t in Periods)
  definition : if ( t in Periods.past) then
                "past"
              elseif ( t in Periods.beyond ) then
                "beyond"
              else
                FormatString("%e",WeekInPeriod(t))
              endif

```

The predefined function `FormatString` allows you to compose a string that is built up from a combination of numbers, strings and set elements (see Chapter 5 of *The Language Reference*).

The above string parameter `PeriodDescription(t)` can be used as element text in the table after executing the following steps:

... as part of the table

- ▶ open the **Table Properties** dialog box of the table,
- ▶ select the **Element Text** tab (see Figure 11.6),
- ▶ select the index `t`,
- ▶ press the **Modify** button,
- ▶ select the identifier `PeriodDescription(t)`,
- ▶ press the **Next** button,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

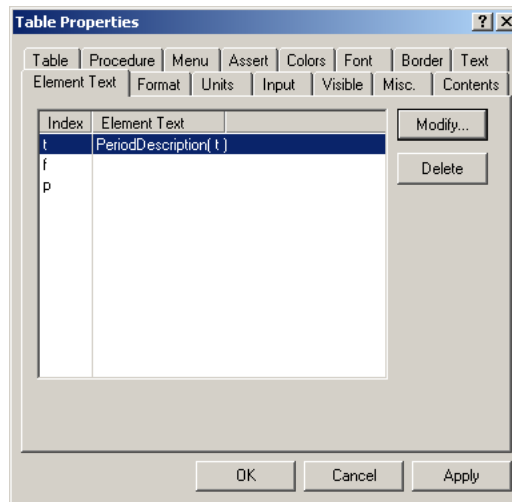


Figure 11.6: The **Element Text** tab of the **Table Properties** dialog box

If the table does not show the constructed period descriptions, and you receive an initialization warning, you should press the **Run Next** button once and the period descriptions should then appear.

Viewing the result

AIMMS chooses a default number format when displaying identifiers in a table. However, you might want to change the number of digits and/or the number of decimals. For example, the parameter `ActualNumberOfDaysInPeriod` should be an integer, and the values of the parameter `ActualProduction` are too large for the default format.

Specifying the number format ...

You can execute the following steps to change the number format of `ActualProduction` to a width of 8 digits with 2 decimals:

... first for actual production

- ▶ open the **Table Properties** dialog box of the table,
- ▶ select the **Format** tab (see Figure 11.7),
- ▶ select the element `ActualProduction(f,p,t)` from the drop-down listbox,
- ▶ enter the number '8' (without quotes) in the 'Width' field,
- ▶ enter the number '2' (without quotes) in the 'Decimals' field, and
- ▶ press the **Apply** button.

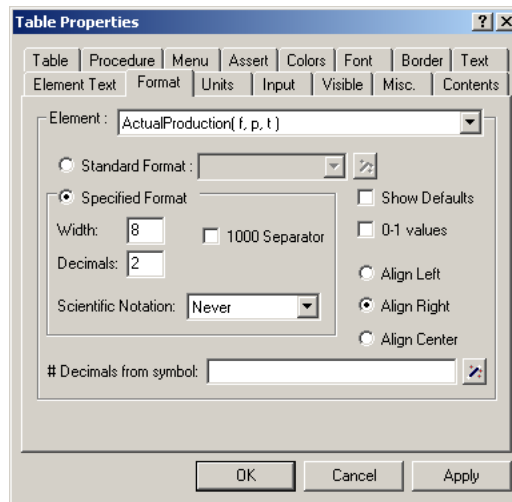


Figure 11.7: The **Format** tab of the **Table Properties** dialog box

Next, you should change the format of the parameter DeteriorationLevel to a width of 5 with 2 decimals, and also adjust the number format of the parameter ActualNumberOfDaysInPeriod to a width of 5 with 0 decimals. An instance of the completed table is shown in Figure 11.8.

... and then for the other two identifiers


	Deterioration	past	Production				
			week 27, 2000	week 28, 2000	week 29, 2000	week 30, 2000	week 31, 2000
Eindhoven							
line-01	3.25		2250.00	2250.00	2250.00	2250.00	2250.00
line-02	6.25						
line-03							
line-04							
Haarlem							
line-01	5.00		2250.00	2250.00	2250.00	2250.00	2250.00
line-02	13.25		2500.00	2500.00	2500.00	2500.00	2500.00
line-03	3.00						
line-04	2.25						
Zwolle							
line-01	6.75		2750.00	2750.00	2750.00	2750.00	2750.00
line-02	6.00						
line-03							
line-04							
Number of Working Days			5	5	5	5	5

Figure 11.8: The completed production line table

11.2.3 The factory production bar chart

The production lines table displays a production overview for each individual production line. The following bar chart will provide a similar overview at the factory level. To create this bar chart you should perform the following actions:

Creating a bar chart

- ▶ make sure that the *Production Overview* page is opened in **Edit** mode,
- ▶ press the **New Bar Chart** button  on the toolbar,
- ▶ drag and create a rectangle underneath the *Production Lines* table with the same width, and
- ▶ select the variable $\text{Production}(f, t)$ using the **Identifier** wizard.

As before, you should change the abstract period references into week references using the string parameter *PeriodDescription*. The resulting bar chart is shown in Figure 11.9.

Creating week labels

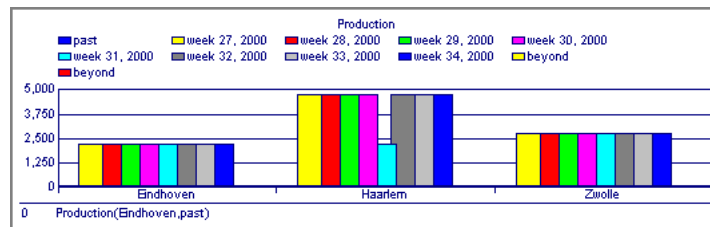



Figure 11.9: The completed factory production bar chart

11.2.4 The vacation table

The created table will display all the weeks that correspond to a vacation period with a 40% drop in production. To create this table you should complete the following sequence of steps:

Creating the table

- ▶ make sure that the *Production Overview* page is in **Edit** mode,
- ▶ press the **New Table** button  on the toolbar,
- ▶ drag and create a rectangle below the factory production bar chart with the same dimensions,
- ▶ select the parameter $\text{IsVacationPeriod}(f, t)$ using the **Identifier** wizard, and
- ▶ change the element text of the index t to the string parameter *PeriodDescription*(t).

The identifier $\text{IsVacationPeriod}(f, t)$ is a binary parameter. A value of zero means 'no vacation period', while a value of one indicates a 'vacation period'. The chosen value of one is somewhat arbitrary, and for this reason you might prefer to display a cross instead of a one. This minor modification can be accomplished as follows:

Displaying nonzero values as crosses

- ▶ open the **Table Properties** dialog box of the table,
- ▶ select the **Format** tab (see Figure 11.10),
- ▶ check the '0-1 values' check box, and

- ▶ press the **OK** button.

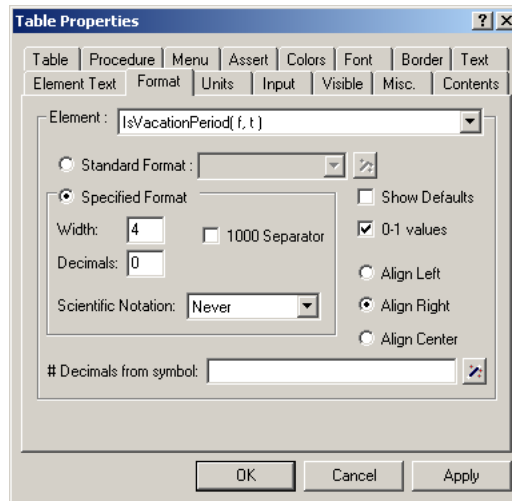



Figure 11.10: The **Format** tab of the **Table Properties** dialog box

Note that at this point the table is still empty since no vacation weeks have yet been specified. Later, you will specify these vacation weeks using a Gantt chart object on the *Absentee Overview* page.

11.2.5 The horizon-calendar tables

In this subsection you will create two composite tables that establish the relationship between the abstract horizon periods and the weekly and daily calendar periods. Composite tables in AIMMS resemble the structure of relational database tables, and you can adjust the width of columns from within the graphical interface. To create your first composite table, you should execute the following steps:

Creating your first composite table

- ▶ press the **New Composite Table** button  on the toolbar,
- ▶ draw a rectangle on the page,
- ▶ select the parameter `WeekInPeriod(t)`,
- ▶ press the **Next** button, and
- ▶ press the **Finish** button.

For the second composite table you should select the indexed set DaysInPeriod(t). The two composite tables should look similar to the ones shown in Figure 11.11.

Creating the second table

t	WeekInPeriod	t	Days
past	week 26, 2000	period-01	03/07/2000
period-01	week 27, 2000	period-01	04/07/2000
period-02	week 28, 2000	period-01	05/07/2000
period-03	week 29, 2000	period-01	06/07/2000
period-04	week 30, 2000	period-01	07/07/2000
period-05	week 31, 2000	period-02	10/07/2000
period-06	week 32, 2000	period-02	11/07/2000
period-07	week 33, 2000	period-02	12/07/2000
period-08	week 34, 2000	period-02	13/07/2000
period-09	week 35, 2000	period-02	14/07/2000
period-10	week 36, 2000	period-03	17/07/2000
		period-03	18/07/2000
		period-03	19/07/2000

Figure 11.11: The mapping between horizon and calendars

11.2.6 The maintenance and mode switches tables

As with vacation periods and holidays, maintenance periods also cause a decrease in production. Therefore, a maintenance overview can also contribute to the interpretation of the results in the production line table and factory production bar chart. By now you should be able to create the maintenance table without guidance. This composite table needs only the identifier LineInMaintenance(f, p, t) as its domain, and the table will immediately contain the required three columns. To complete the table you should again change the abstract period references by specifying that the string parameter PeriodDescription(t) is used as the element text of the index t (as you did previously).

Creating the maintenance table

The last composite table on the *Production Overview* page will display all the optimal mode switches for the current planning horizon. It can be specified in the same way as the table in the previous paragraph. The identifier ProductionLineLevelChange(f, p, t) is used to specify the domain of the table. The two composite tables are shown in Figure 11.12.

Creating the mode switch table

f	p	t	f	p	t
Haarlem	line-02	week 31, 2000	Eindhoven	line-02	week 27, 2000
			Haarlem	line-04	week 27, 2000

Figure 11.12: The maintenance (left) and mode switch (right) tables

11.2.7 The total costs bar chart

The final data object on this page will display the four cost components that together determine the overall total cost, in an aggregated way. As of yet, there are no identifiers that contain the values of these four components. Therefore, you must first declare four new parameters describing the aggregated production, transport, stock and mode-switch costs which are to be placed at the end of the Production Overview section. Note that the aggregated transport and stock costs are expected costs.

*Declaring
auxiliary
parameters*

```

PARAMETER:
  identifier : TotalProductionCost
  unit      : $
  definition : sum[ (f,t), UnitProductionCost(f) * Production(f,t) ]

PARAMETER:
  identifier : TotalTransportCost
  unit      : $
  definition : sum[ (f,c,t,s), ScenarioProbability(s) * UnitTransportCost(f,c) *
                  Transport(f,c,t,s) ]

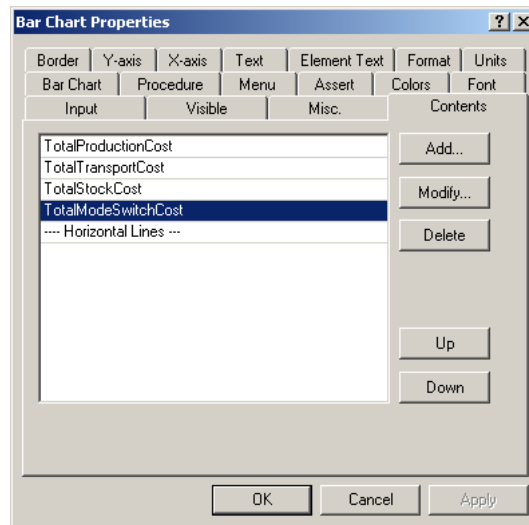
PARAMETER:
  identifier : TotalStockCost
  unit      : $
  definition : sum[ (l,t,s), ScenarioProbability(s) * UnitStockCost(l) *
                  Stock(l,t,s) ]

PARAMETER:
  identifier : TotalModeSwitchCost
  unit      : $
  definition : sum[ (f,p,t), FixedCostDueToLevelChange *
                  ProductionLineLevelChange(f,p,t) ]

```

Following the declaration of the above four identifiers, you can now create a bar chart object with as its first identifier TotalProductionCost. You can then open the **Bar Chart Properties** dialog box and use the **Contents** tab to add the remaining three identifiers (see Figure 11.13). You can ignore all the initialization warnings.

*Creating a bar
chart*

Figure 11.13: The **Contents** tab of the **Bar Chart Properties** dialog box

The completed total costs bar chart should look like the one shown in Figure 11.14.

Viewing the result

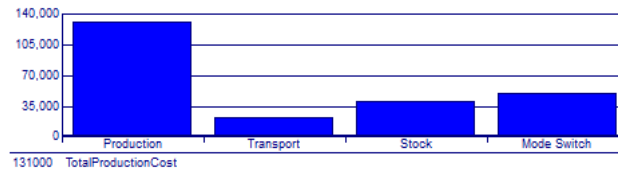



Figure 11.14: The completed total cost bar chart

11.2.8 Completing the page

One way to display more information within objects on a page is to reduce the size of the font used. To create a new, small, font for use with all data objects you should execute the following actions:

Changing fonts

- ▶ make sure that the *Production Overview* page is in **Edit** mode,
- ▶ select a table, and then
- ▶ select the remaining seven tables and bar charts while keeping the *Shift* key pressed,
- ▶ press the **Properties** button  on the toolbar,
- ▶ select the **Font** tab, and
- ▶ press the **Add** button,
- ▶ enter '7' as the 'Font Size' (see Figure 11.15),

- ▶ press the **OK** button,
- ▶ specify 'Data Font' as the name of the new font, and
- ▶ press the **OK** button twice.

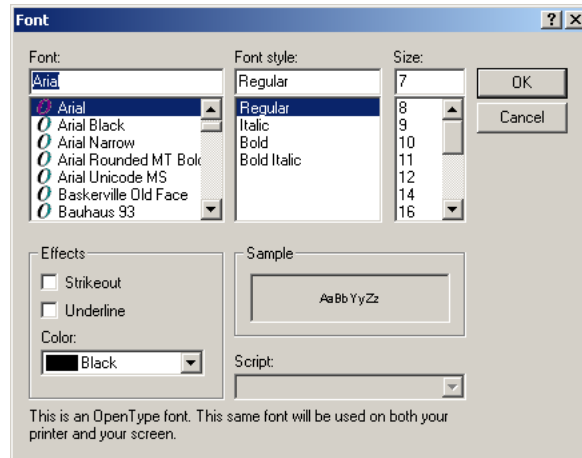


Figure 11.15: The specification of a new font

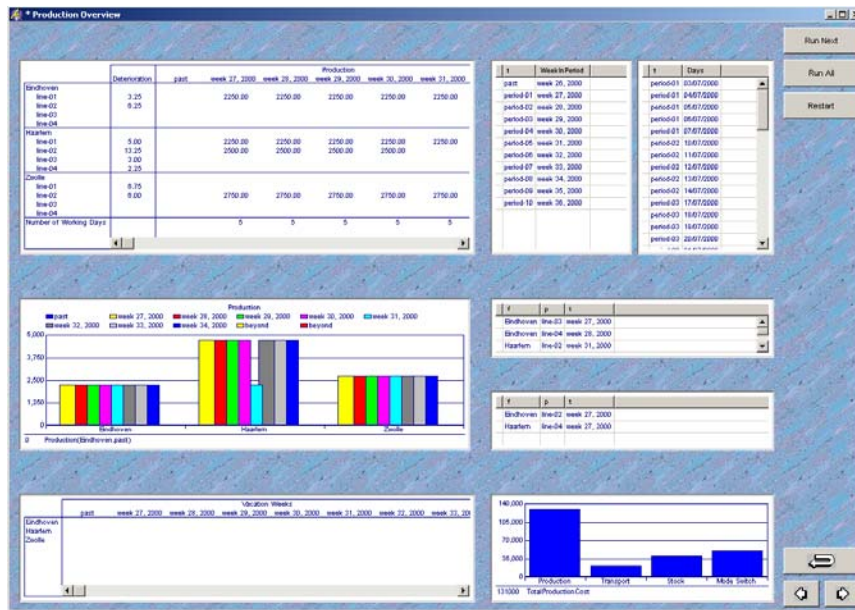
Several tables, bar charts and composite tables have been placed on the *Production Overview* page. To complete the page you should first align and resize the page objects in order to create a structured and attractive composition. For this purpose AIMMS offers several alignment tools that are accessible through the **Alignment** submenu of the **Edit** menu. The following alignment options are supported:

- aligning objects to the *left*, *right*, *top* or *bottom*,
- centering objects *horizontally* or *vertically*,
- spreading objects *horizontally* or *vertically*, and
- making object size equal in *width* or *height*.

You should now use the alignment tools described in the previous paragraph to align all the page objects as shown in Figure 11.16. Remember, if you need to select several objects at once, you should keep the *Shift* key pressed.

Alignment of objects

Aligning the Production Overview page

Figure 11.16: Aligned objects on the *Production Overview* page

Adding text to objects will help the end-user of your application. In this paragraph you will create a text object, and in the next paragraph you will change the font associated with this text. Consider first the production line table in the upper left corner, and add a line of text by following these steps:

Creating the text objects

- ▶ make sure that the *Production Overview* page is in **Edit** mode,
- ▶ select the **Text** command from the **Object** menu,
- ▶ draw a rectangle above the production line table,
- ▶ enter 'Production Lines' (without quotes) in the edit field (see also Figure 11.17), and
- ▶ press the **OK** button.

You should now create six more text objects as shown in Figure 11.2 at the beginning of this chapter.

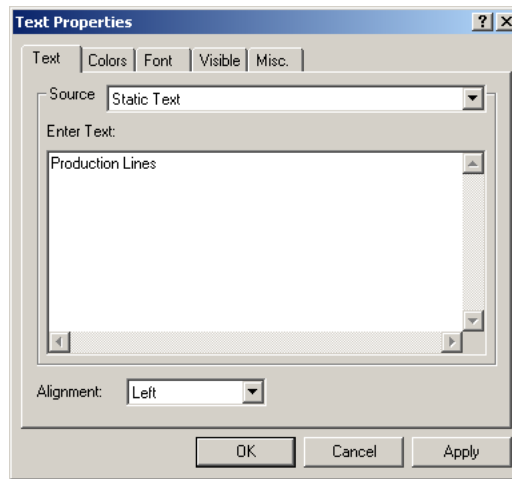


Figure 11.17: The **Text** tab of the **Text Properties** dialog box

To change the font size of the text objects referred to in the previous paragraph, first select all of them using the *Shift* key, and create a new font named 'Title Font' with 'Font Size' 18. Again, you are referred to the text objects as shown in Figure 11.2.

Changing the text font

To improve the structure of your page even further, you can enclose one or more page objects within a rectangle. The following steps are required:


Creating the rectangles

- ▶ make sure that the *Production Overview* page is in **Edit** mode,
- ▶ select the **Rectangle** command from the **Object** menu, and
- ▶ draw the rectangle around an object on your page.

Again, you should try to match the six rectangles as shown in Figure 11.2.

To embolden your rectangles you can enlarge their line thickness by executing the following actions:

Rectangle line size

- ▶ make sure that the *Production Overview* page is in **Edit** mode,
- ▶ select all rectangles using the *Shift* key,
- ▶ press the **Properties** button  on the toolbar,
- ▶ complete the **Rectangle** tab of the **Rectangle Properties** dialog box as shown in Figure 11.18, and
- ▶ press the **OK** button.

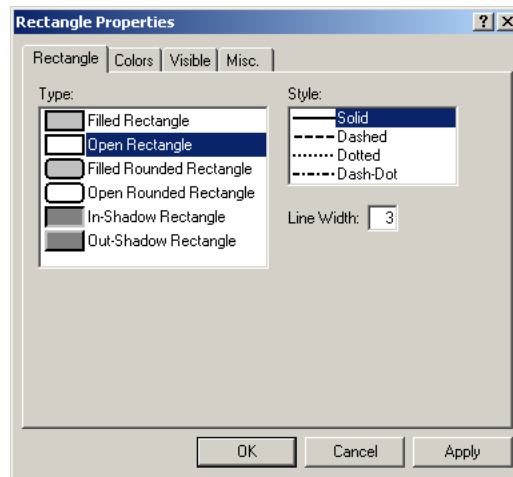



Figure 11.18: The **Rectangle** tab of the **Rectangle Properties** dialog box

To change the default foreground color of all objects on the page from black to navy blue, you need to execute the following steps:

Changing the foreground color

- ▶ make sure that the *Production Overview* page is in **Edit** mode,
- ▶ press the *Ctrl-A* key combination to select all objects on the page,
- ▶ unselect the three execution button using the *Shift* key,
- ▶ press the **Properties** button  on the toolbar,
- ▶ select the **Colors** tab,
- ▶ select 'User' as the determinant of the 'Foreground' color,
- ▶ select the color 'Navy Blue' from the drop-down list, and
- ▶ press the **OK** button.

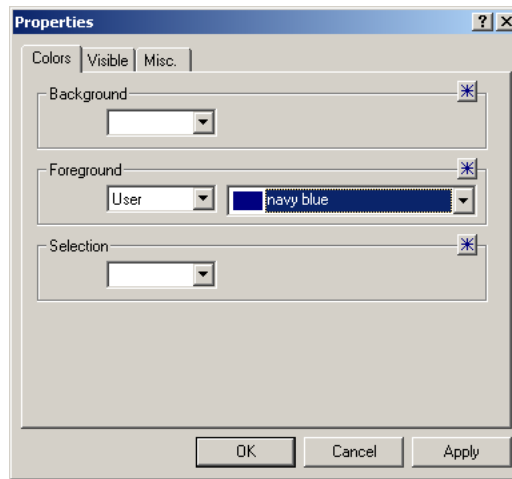
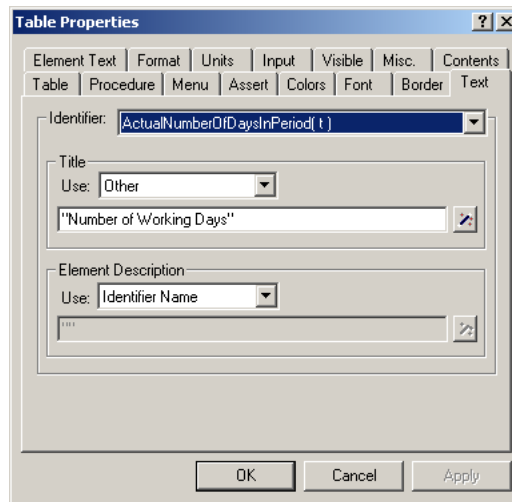


Figure 11.19: The **Colors** tab of the **Properties** dialog box

By default, AIMMS will display the identifier names inside data objects. If this default name needs to be changed for your end-user, you can enter your own preferred string. You can even enter a string parameter, so that you can serve end-users with different language needs. As an illustration, please change the default representation of the identifier `ActualNumberOfDaysInPeriod` to the string 'Number of working days' by performing the following steps:

*Changing text
inside objects*

- ▶ select the production lines table,
- ▶ open its **Table Properties** dialog box,
- ▶ select the **Text** tab,
- ▶ select the identifier `ActualNumberOfDaysInPeriod(t)`,
- ▶ select 'Other' from the drop-down list in the 'Title' section,
- ▶ specify "Number of working days" (in quotes) as the new title (see Figure 11.20), and
- ▶ press the **OK** button.

Figure 11.20: The **Text** tab of the **Table Properties** dialog box

In AIMMS it is even possible to color the individual data entries in tables. For instance, you might want to display the deterioration levels in red instead of blue whenever these levels have reached their maximum. To do this, you should first create a so-called *color parameter*. Such a parameter is an element parameter in the predefined AIMMS set `AllColors`. The contents of this set can be inspected or changed using the **User Colors** command from the **Tools** menu.

Coloring data entries

As an example, please declare the following color parameter in the Production Overview Declarations section:


Creating a color parameter ...

```
ELEMENT PARAMETER:
  identifier : DeteriorationColor
  index domain : (f,p) | p in FactoryProductionLines(f)
  range      : AllColors
  definition  : if ( DeteriorationLevel(f,p) > MaximumDeteriorationLevel(f,p) ) then
                'red'
              else
                'navy blue'
              endif
```

To specify the actual link between the color parameter and the data in the table you should perform the following actions:

... and linking it to model data

- ▶ open the **Table Properties** dialog box of the production lines table,
- ▶ select the **Colors** tab,
- ▶ select the identifier `DeteriorationLevel(f,p)` in the 'Identifier' section (at the bottom),

- ▶ select 'Model' as the color determiner,
- ▶ press the **Wizard** button  (see Figure 11.21) to select the identifier DeteriorationColor(f,p), and
- ▶ press the **OK** button.

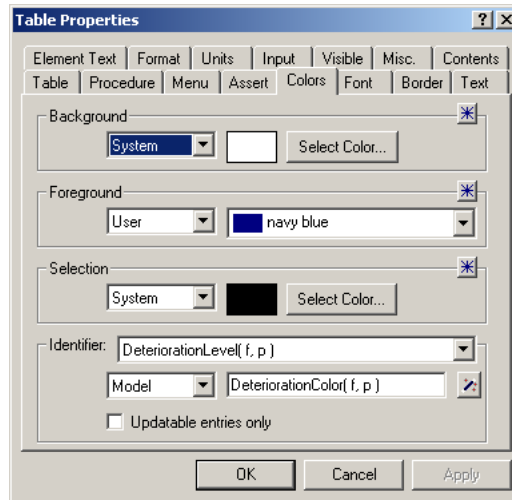
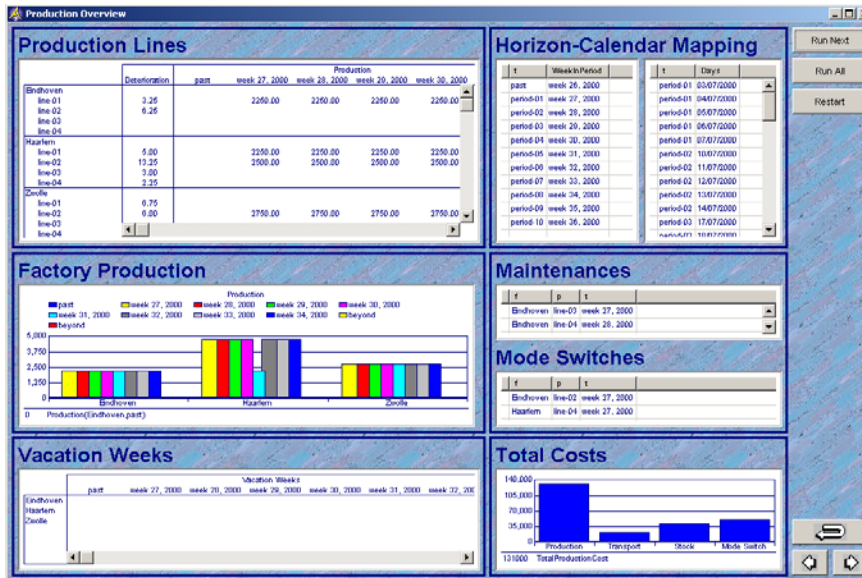


Figure 11.21: The **Colors** tab of the **Table properties** dialog box

The completed *Production Overview* page is repeated in Figure 11.22, so that you can compare it with the contents of your screen.

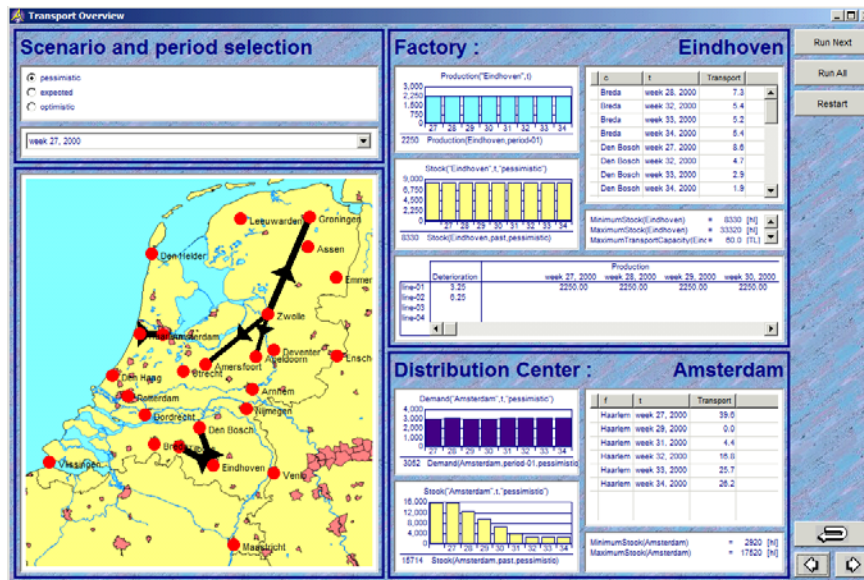
The completed page

Figure 11.22: The completed *Production Overview* page

11.3 The Transport Overview page

In this section you will construct the entire *Transport Overview* page as shown in Figure 11.23. Each page object is covered by a separate subsection.

Viewing the entire page

Figure 11.23: The completed *Transport Overview* page

11.3.1 Scenario selection object

The values of the identifiers *Transport* and *Stock* are different for each demand scenario. Displaying these values for all scenarios on a single page would overload the page. Therefore, the displayed information will be limited to one scenario, and the end-user will be able to switch between scenarios. AIMMS provides a *selection object* for this purpose.

Scenario dependency

In the model section *Transport Overview* you should first create a new declaration section *Transport Overview Declarations* containing the following element parameter:


Creating a scenario parameter

```
ELEMENT PARAMETER:
  identifier : DisplayedScenario
  range     : Scenarios
```

The value of this element parameter is then determined by linking it to a selection object through the following steps:

Creating a selection object

- ▶ open the *Transport Overview* page in **Edit** mode,
- ▶ press the **New Selection Object** button  on the toolbar,

- ▶ drag and create a small rectangle in the upper left corner,
- ▶ select 'Radio Buttons' from the 'Single Item Selection' options,
- ▶ select 'Element Parameter' as the 'Type of Data',
- ▶ press the **Wizard** button  next to the 'Element' field (see Figure 11.24),
- ▶ select the element parameter DisplayedScenario,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

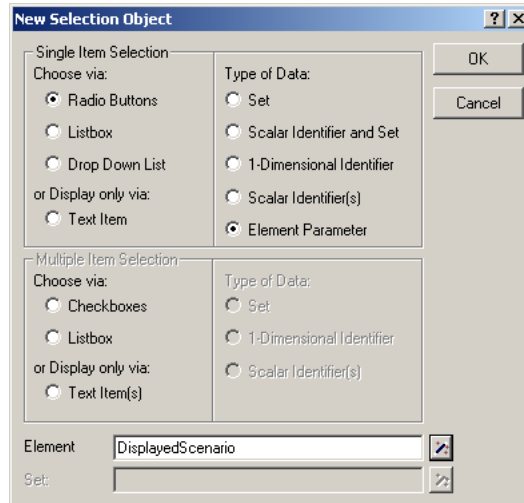


Figure 11.24: The **New Selection Object** dialog box

The selection object that you have created is shown in Figure 11.25. Selecting a radio button in the selection object will set the corresponding value of the element parameter `DisplayedScenario`. As you will see later in this section, other page objects will be defined over this element parameter, and their data will adjust accordingly.

Using the selection object

- pessimistic
- expected
- optimistic

Figure 11.25: The scenario selection object

11.3.2 Period selection object

As with the element parameter `DisplayedScenario`, you can introduce another element parameter to support the selection of a particular period. Please declare the following element parameter at the end of the section `Transport Overview`.

Creating a period parameter

```
ELEMENT PARAMETER:
  identifier : DisplayedPeriod
  range     : Periods
```

When creating the selection object that sets the element parameter `DisplayedPeriod`, you should select the ‘Drop Down List’ option rather than the ‘Radio Buttons’ option (see Figure 11.26).

Creating the drop down list

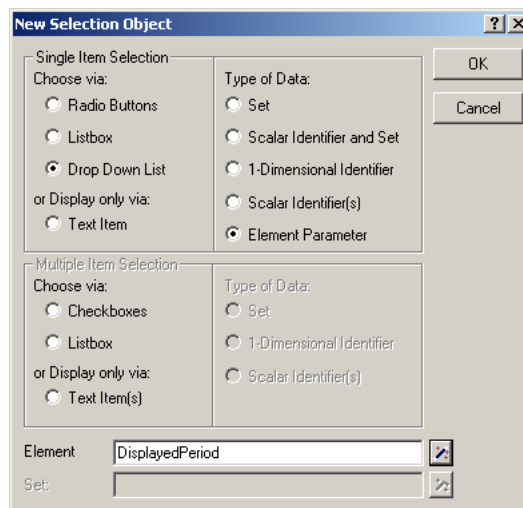


Figure 11.26: The **New Selection Object** dialog box

Once you have created the drop down list, you can open its **Selection Object Properties** dialog box (either by double-clicking or using the right-mouse pop-up menu), and change the element text from abstract period references to specific week references. You can accomplish this change by selecting the **Element Text** tab, and specifying the string parameter `PeriodDescription(t)` as the element text of the index `Periods`.

Specifying element text




To initialize the two element parameters `DisplayedScenario` and `DisplayedPeriod` you should temporarily change the page mode to **User** mode, and use the two selection objects to select 'optimistic' as the displayed scenario and 'week 27, 2000' as the displayed period.

*Initializing
element
parameters*

11.3.3 Transport network object

The third object to be created on the transport page is a network object displaying the optimal transports for a given scenario and a given period in the planning interval. In Chapter ?? you created a network object displaying all locations and this will be used to create the new network object. To copy the existing network from the *Locations* page to the *Transport Overview* page you should perform the following steps:

*Copying the
network object*

- ▶ open both the *Locations* and the *Transport Overview* pages in **Edit** mode,
- ▶ select the *Locations* page tab,
- ▶ select the network object on the *Locations* page,
- ▶ press the **Copy** button  on the toolbar,
- ▶ close the page by clicking on the cross  in the upper right corner,
- ▶ select the *Transport Overview* page tab,
- ▶ press the **Paste** button ,
- ▶ position the network object underneath the selection object, and
- ▶ press the left-mouse button.

The network object that you created in Chapter ?? only showed the locations. You can now add arcs to the network object to represent the optimal transport between the factories and the distribution centers for a given period and a given scenario. To add these arcs, you should take the following actions:

*Adding arcs to
the network*

- ▶ select the network object in **Edit** mode,
- ▶ open its **Network Object Properties** dialog box,
- ▶ select the **Contents** tab,
- ▶ select the '----- Arcs -----' entry from the listbox,
- ▶ press the **Add** button,
- ▶ select the variable `Transport(f,c,t,s)`, and
- ▶ press the **Next** button.

Next you need to specify that the indices `t` and `s` will assume the values of the element parameters `DisplayedPeriod` and `DisplayedScenario` respectively:

- ▶ select the index `t` from the list box,
- ▶ select the 'Element Parameter' radio button,
- ▶ select the element parameter `DisplayedPeriod` from the drop-down list,
- ▶ select the index `s` from the list box,
- ▶ select the 'Element Parameter' radio button,
- ▶ select the element parameter `DisplayedScenario` from the drop-down list,

- ▶ press the **Finish** button (see Figure 11.27), and
- ▶ press the **OK** button.

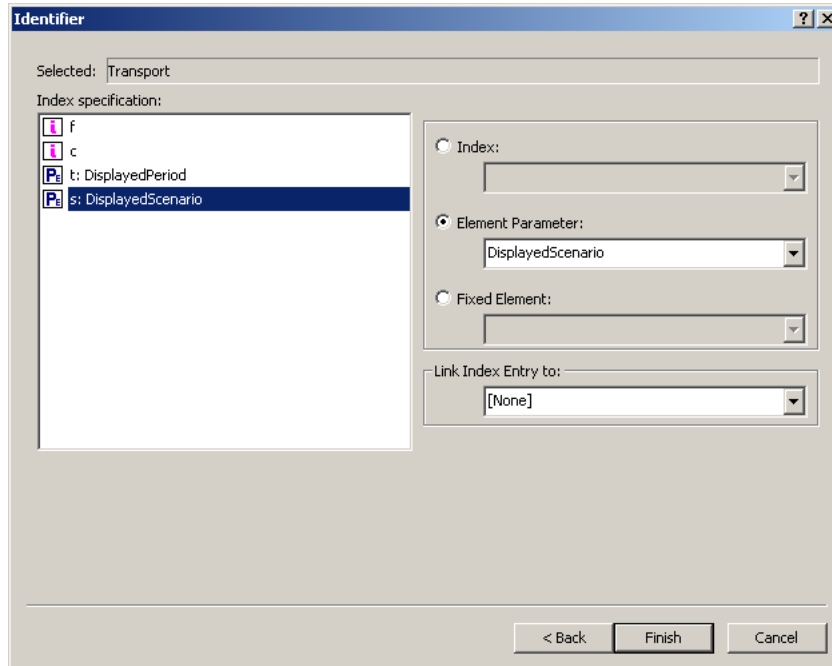



Figure 11.27: Fixing indices of the variable Transport

The network object will display arcs for all transport values that have a non-zero value. To distinguish between small and large transport values the thickness of the arc can be varied depending on the transport value. To achieve this you should execute the following actions:

Specifying arc thickness

- ▶ select the network object in **Edit** mode,
- ▶ open its **Network Object Properties** dialog box,
- ▶ select the **Arcs** tab,
- ▶ press the **Wizard** button  to the right of the 'Size' field,
- ▶ select the identifier Transport(f,c,t,s),
- ▶ press the **Next** button,
- ▶ link the index t to the element parameter DisplayedPeriod,
- ▶ link the index s to the element parameter DisplayedScenario,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

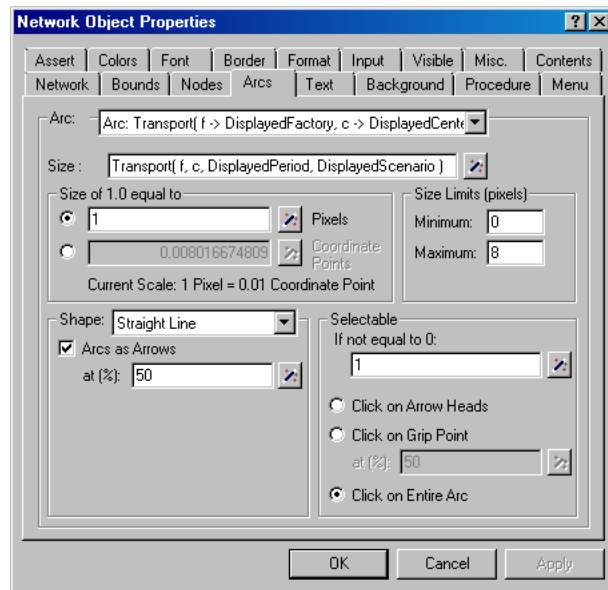


Figure 11.28: The Arcs tab of the Network Properties dialog box

Assuming that you have already solved the model for the first step, the arcs in the network object should now have different widths as shown in Figure 11.29.

Viewing the arcs

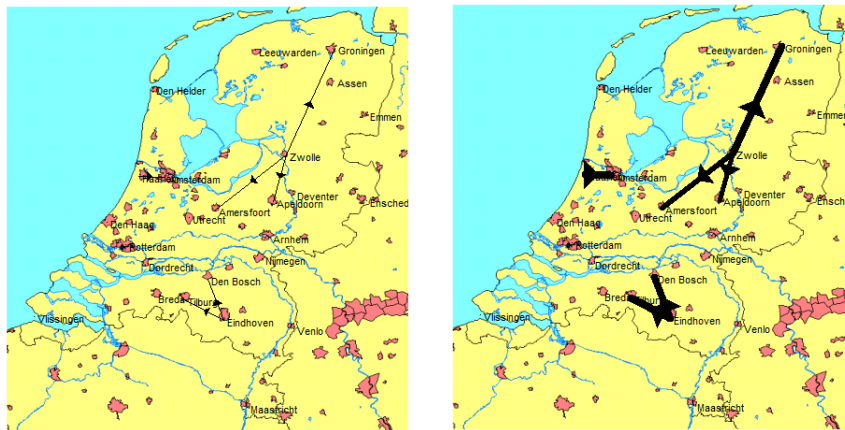


Figure 11.29: Using arc thickness to illustrate transport volumes

AIMMS has facilities to display node and arc dependent information whenever an end-user selects a node or an arc in the network object. Consider, for instance, Figure 11.23. The data block displayed in the lower right corner of that page deals with a particular distribution center, while the data block

Node and arc dependent information

displayed in the upper right corner deals with a particular factory. In the following paragraphs you will specify how the selection of a particular arc will update both of these data blocks, while the selection of a particular node will update one of these data blocks.

The following two element parameters will be needed to hold the current choice of factory and distribution center. Please add their declarations to the Transport Overview Declarations.

*Declaring
location
identifiers*

```
ELEMENT PARAMETER:
  identifier  : DisplayedFactory
  range      : Factories

ELEMENT PARAMETER:
  identifier  : DisplayedCenter
  range      : Centers
```

Arc dependency can then be specified with the aid of the above two element parameters. Whenever an arc is selected, the locations of the corresponding two end nodes should become the current values of `DisplayedFactory` and `DisplayedCenter`. As soon as their values change, the data blocks in Figure 11.23 will be updated accordingly. To implement this action, you should execute the following steps:

*Specifying arc
dependency*

- ▶ select the network object in **Edit** mode,
- ▶ open its **Network Object Properties** dialog box,
- ▶ select the **Contents** tab,
- ▶ select the arc `Transport(f,c,DisplayedPeriod,DisplayedScenario)`,
- ▶ press the **Modify** button,
- ▶ press the **Next** button,
- ▶ select the index `f` from the 'Index specification' list box,
- ▶ use the drop-down list under 'Link Index Entry' to select the element parameter `DisplayedFactory`,
- ▶ repeat the previous two steps to link the index `c` the element parameter `DisplayedCenter`,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

By simply linking an index to an element parameter as shown in Figure 11.30 you have specified the linkage between a selection and a data block. This powerful facility is also available for other data objects in AIMMS.

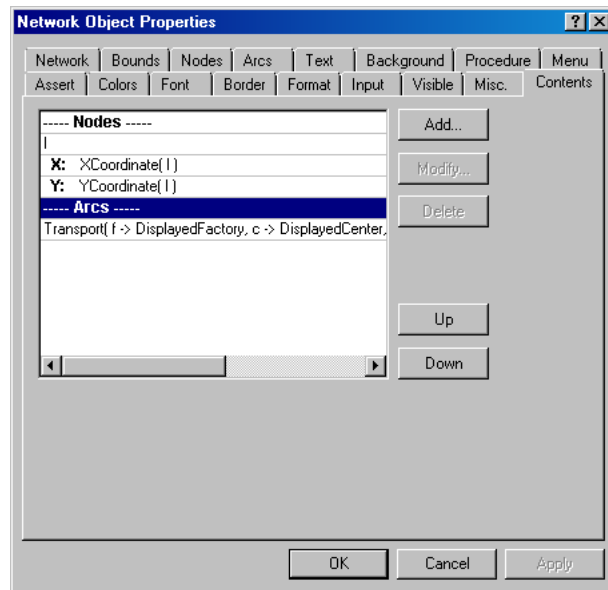


Figure 11.30: The **Contents** tab of the **Network Object Properties** dialog box

Specifying node dependency is not as straightforward as with arc dependency, because a node is a location that can be either a factory or a distribution center. This makes the linkage between a node and one of the data blocks less trivial to specify. A straightforward procedure, however, can resolve this choice. Once you have specified such a procedure, it is then straightforward to link it to the network object.

Specifying node dependency

Create a procedure `SelectLocationInNetwork(SelectedLocation)`, where the argument `SelectedLocation` is declared as a local element parameter with **Range** attribute `Locations` and with the **Property** attribute 'Input' as shown in Figure 11.31. The following conditional statement will constitute the **Body** attribute of this procedure:

The selection procedure ...

```
if ( SelectedLocation in Factories )
  then DisplayedFactory := SelectedLocation ;
  else DisplayedCenter := SelectedLocation ;
endif;
```

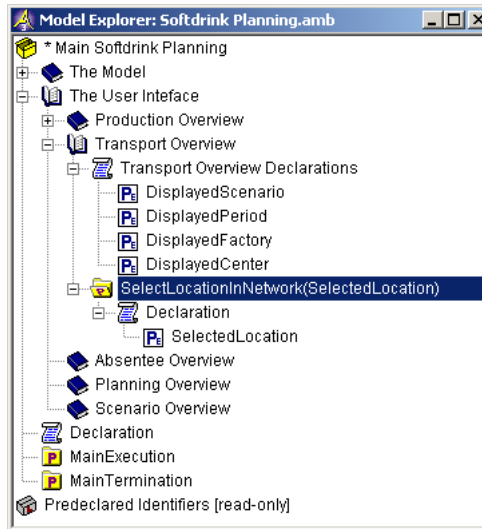


Figure 11.31: The contents of the Transport Overview section

The above procedure will be linked to the network object as a *procedure upon selection* by executing the following steps: *... linked to the network object*

- ▶ select the network object in **Edit** mode,
- ▶ open its **Network Properties** dialog box,
- ▶ select the **Procedure** tab,
- ▶ verify that 'Node: l' is selected as the 'Identifier',
- ▶ select the procedure `SelectLocationInNetwork` as the 'Upon Selection' procedure,
- ▶ press the **Next** button,
- ▶ select the 'Index' radio button,
- ▶ select the index 1 from the 'Index' drop-down list (see Figure 11.32),
- ▶ press the **Finish** key, and
- ▶ press the **OK** button.

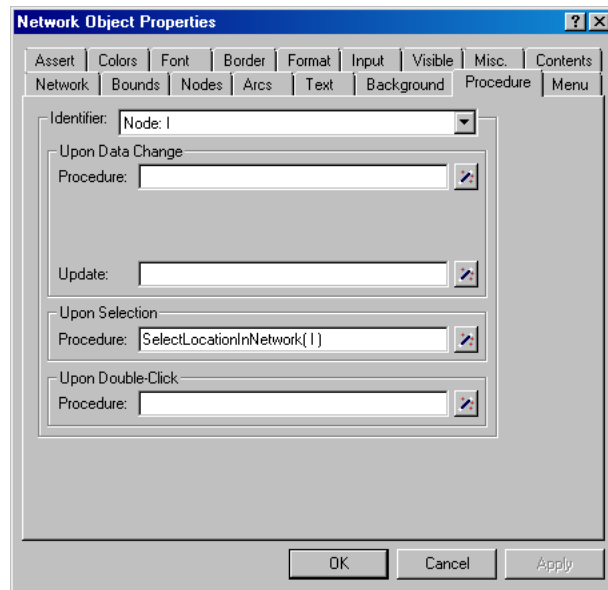


Figure 11.32: The **Procedure** tab of **Network Object Properties** dialog box

In order to see nodes in the network more clearly, you can increase their size by changing the **Nodes** tab of the network object as shown in Figure 11.33. If you want, you can also change their color using the **Colors** tab.

Increasing the node size

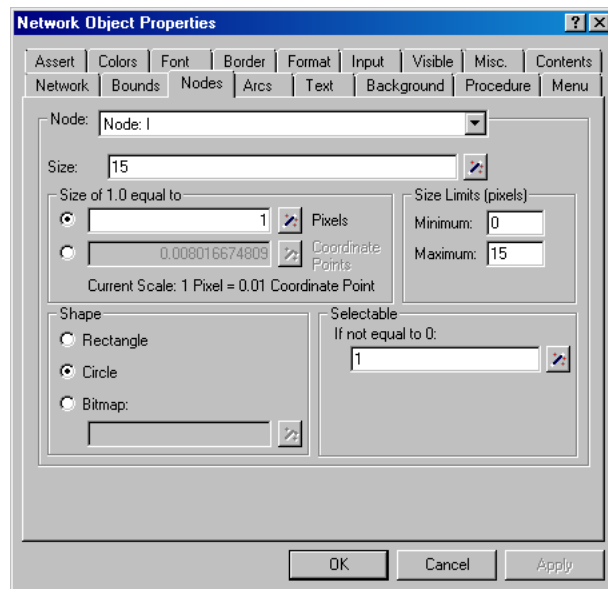


Figure 11.33: The **Nodes** tab of the **Network Object Properties** dialog box

Once you have increased the node size, the network object should look like the one shown in Figure 11.34. If you had used separate node sets for factories and centers, different icons could have been used to represent them in the network object.

Viewing the result



Figure 11.34: The network object with increased node size

11.3.4 Factory text object

The upper right data block in the *Transport Overview* page contains data pertaining to a particular factory. The name of that factory is displayed at the top of this block using a text object. The following string parameter is needed to fill this object:

Factory description

```
STRING PARAMETER:
  identifier : FactoryDescription
  definition : FormatString( "%e", DisplayedFactory )
```

You should add this declaration at the end of the *Transport Overview* Declarations section.

You should now create a text object that will display the contents of the string parameter you have just declared. Try to create the text object on your own. To display the string parameter `FactoryDescription` you should complete the **Text** tab of the **Text Properties** dialog box as shown in Figure 11.35. You can also try changing its color and font size.

Creating a text object

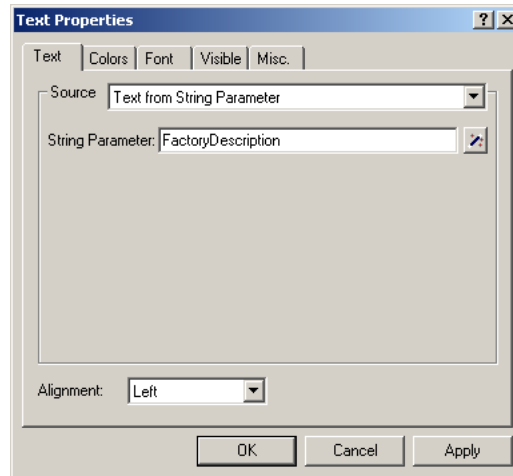



Figure 11.35: The **Text** tab of the **Text Properties** dialog box

11.3.5 The factory production bar chart

You will begin by creating a bar chart containing the production data corresponding to the currently selected factory. The name of this factory is the value of the element parameter `DisplayedFactory`. You should execute the following steps:

Creating the bar chart

- ▶ make sure the *Transport Overview* page is opened in **Edit** mode,
- ▶ press the **New Bar Chart** button  on the toolbar,
- ▶ drag a rectangle underneath the factory description text object,
- ▶ select the variable `Production(f,t)` in the **Identifier** wizard,
- ▶ press the **Next** button,
- ▶ link the index `f` to the element parameter `DisplayedFactory`, and
- ▶ press the **Finish** button.

The period references along the x -axis are probably too long to fit. The `Period-Description` parameter contains even longer strings. To create short references you should now create the following string parameter:

Adjusting the element text

```

STRING PARAMETER:
  identifier : ShortPeriodDescription
  index domain : t
  definition :
    if ( WeekInPeriod(t) )
      then FormatString( "%n", TimeslotCharacteristic( WeekInPeriod(t), 'week' ) )
    else ""
    endif

```

You should change the element description of the period index *t* to be the string parameter *ShortPeriodDescription* using the **Element text** tab of the **Bar Chart Properties** dialog box.

At this point, the page on your screen should resemble the partially completed *Transport Overview* page shown in Figure 11.36. *The page so far*

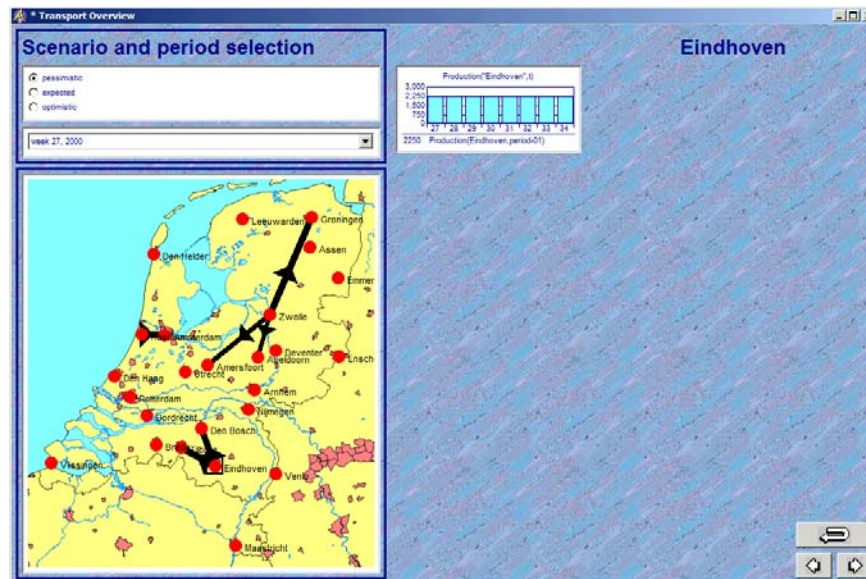





Figure 11.36: The current *Transport Overview* page

11.3.6 The factory stock bar chart

To create a bar chart containing the stock values for the currently selected factory, you can make use of the following copy, paste and adjust actions:

- ▶ select the production bar chart you have just created,
- ▶ press the **Copy** button  on the toolbar,


Copying the previous bar chart

- ▶ press the **Paste** button  on the toolbar,
- ▶ position and drop the new bar chart underneath the production bar chart,
- ▶ press the **Properties** button  on the toolbar,
- ▶ select the **Contents** tab,
- ▶ select the identifier Production(DisplayedFactory, t) from the listbox,
- ▶ press the **Modify** button,
- ▶ select the identifier Stock(l, t, s),
- ▶ press the **Next** button,
- ▶ link the index l to the element parameter DisplayedFactory,
- ▶ link the index s to the element parameter DisplayedScenario,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

11.3.7 Factory transport composite table

The network object only displays transport values for the selected period. To view the transport values for all periods in the planning interval you can create a composite table by executing the following steps:

Specifying the table ...

- ▶ press the **New Composite Table** button  on the toolbar,
- ▶ draw a rectangle on the page,
- ▶ select the variable Transport(f, c, t, s),
- ▶ press the **Next** button,
- ▶ link the index f to the element parameter DisplayedFactory,
- ▶ link the index s to the element parameter DisplayedScenario,
- ▶ press the **Finish** button,

You can improve the overall appearance of the table by taking the following actions:

... and improving its appearance

- specify the string parameter PeriodDescription(t) as the element text of the index t, and
- change the font to the 'Data Font' that you specified in Subsection 11.2.8.

The resulting table should now look like the one shown in Figure 11.37.

c	t	Transport
Breda	week 28, 2000	7.3
Breda	week 32, 2000	5.4
Breda	week 33, 2000	5.2
Breda	week 34, 2000	5.4
Den Bosch	week 27, 2000	8.8
Den Bosch	week 32, 2000	4.7
Den Bosch	week 33, 2000	2.9
Den Bosch	week 34, 2000	1.9

Figure 11.37: The factory transport composite table

11.3.8 Factory properties scalar object

To be able to view the minimum and maximum stock levels as well as the maximum transport capacity for the selected factory, you should first create a scalar object with the first of these identifiers:

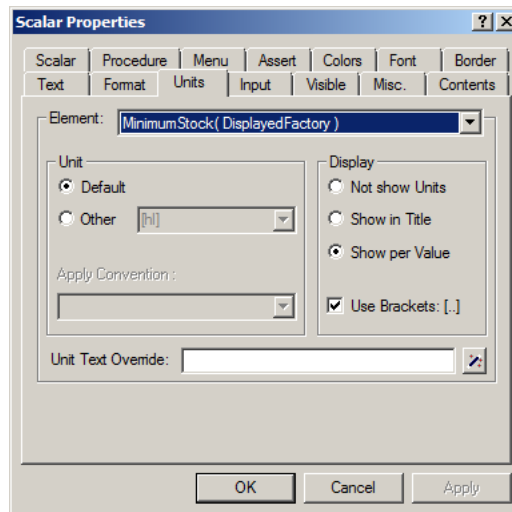
- ▶ create a scalar object,
- ▶ select the identifier `MinimumStock(l)`, and
- ▶ link its index `l` to the element parameter `DisplayedFactory`.

Next, you should add the remaining two identifiers to the scalar object by performing the following actions:

- ▶ open the **Properties** dialog box,
- ▶ select the **Contents** tab,
- ▶ press the **Add** button,
- ▶ select the identifier `MaximumStock(l)`,
- ▶ press the **Next** button,
- ▶ link the index `l` to the element parameter `DisplayedFactory`,
- ▶ press the **Finish** button,
- ▶ press the **Add** button,
- ▶ select the identifier `MaximumTransportCapacity(f)`,
- ▶ press the **Next** button,
- ▶ link the index `f` to the element parameter `DisplayedFactory`,
- ▶ press the **Finish** button, and
- ▶ press the **Apply** button.

Identifier `MinimumStock(l)` and `MaximumStock(l)` have different unit from `MaximumTransportCapacity(f)`. The unit of each identifier will be shown by the following steps:

- ▶ select the **Units** tab of the **Properties** dialog box,
- ▶ it shows the setting of the first identifier `MinimumStock(DisplayedCenter)`,
- ▶ select the **Show per Value** radio button under **Display** as Figure 11.38,
- ▶ click the drop down list on top of the dialog,
- ▶ select the second identifier `MaximumStock(DisplayedCenter)`,
- ▶ again, select the **Show per Value** radio button under **Display**,
- ▶ repeat this for `MaximumTransportCapacity(DisplayedCenter)` as well, and
- ▶ press the **OK** button.

Figure 11.38: The **Property** dialog of factory scalar object

The resulting table should look like the one shown in Figure 11.39 including the appropriate values.



MinimumStock(Eindhoven)	=	8330 [h]
MaximumStock(Eindhoven)	=	33320 [h]
MaximumTransportCapacity(Eindhoven)	=	60.0 [TL]

Figure 11.39: The factory scalar object containing factory limitations

11.3.9 Factory production line table

The factory production line table is essentially the same as the production line table on the *Production Overview* page with the exception that the index *f* is replaced by the element parameter *DisplayedFactory*. The following steps involve copying the table from one page to the next:

Copying the production line table

- ▶ open both the *Production Overview* and the *Transport Overview* page in **Edit** mode,
- ▶ select the *Production Overview* page tab,
- ▶ select the production line table,
- ▶ press the **Copy** button  on the toolbar,
- ▶ close the page,
- ▶ select the *Transport Overview* page tab,
- ▶ press the **Paste** button ,
- ▶ position the object underneath the other factory information objects, and


- ▶ press the left-mouse button.

The following changes are required to display only the information for the currently selected factory:

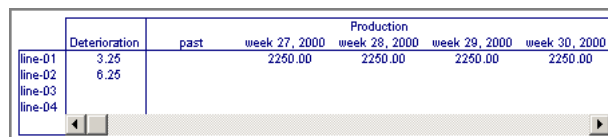
Changing table properties

- ▶ open the **Properties** dialog box of the new table,
- ▶ select the **Contents** tab,
- ▶ select the `DeteriorationLevel(f,p)` entry in the list,
- ▶ press the **Modify** button,
- ▶ press the **Next** button,
- ▶ link the index `f` to the element parameter `DisplayedFactory` and close the wizard,
- ▶ select the `ActualProduction(f,p,t)` entry in the list,
- ▶ press the **Modify** button,
- ▶ press the **Next** button,
- ▶ link the index `f` to the element parameter `DisplayedFactory` and close the wizard,
- ▶ select the `ActualNumberOfDaysInPeriod(t)` entry from the list,
- ▶ press the **Delete** button, and
- ▶ press the **Apply** button.

An error dialog will appear due to the fact that on the **Colors** tab there is still reference to the index `f`. By pressing the **Ok** on the dialog window, AIMMS will get rid of the index reference (i.e. removing the `DeteriorationColor(f,p)`). Therefore, you have to specify color for the `DeteriorationLevel(f,p)` again and change the index reference. This can be done by executing the following steps:

- ▶ select the **Colors** tab,
- ▶ in the 'Identifier' section select 'Model' as the color determiner,
- ▶ press the **Wizard** button  again to select the identifier `DeteriorationColor(f,p)`
- ▶ link the index `f` to the element parameter `DisplayedFactory`,
- ▶ press the **Finish** button, and
- ▶ press the **OK** button.

The resulting table is shown in Figure 11.40.



	Deterioration	Production				
		past	week 27, 2000	week 28, 2000	week 29, 2000	week 30, 2000
line-01	3.25		2260.00	2260.00	2260.00	2260.00
line-02	6.25					
line-03						
line-04						

Figure 11.40: The factory production line table

At this stage you should use the aligning and resizing facilities that were discussed in Subsection 11.2.8 to rearrange the composition objects as shown in Figure 11.41. Once the factory data block is neatly organized, you can copy it in its entirety to create a similar data block for distribution centers.

Arranging the factory objects

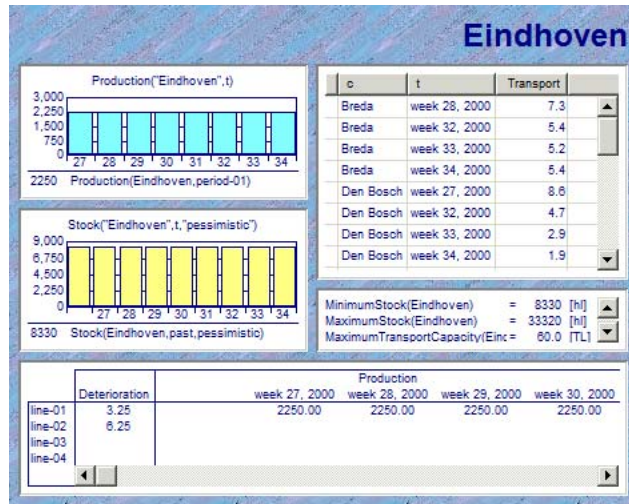




Figure 11.41: The factory data block

11.3.10 The distribution center data block

To create the four page objects for a particular distribution center you should execute the following steps:

- ▶ select all objects in the factory data block except for the production lines table at the bottom using the *Shift* key,
- ▶ press the **Copy** button  on the toolbar,
- ▶ press the **Paste** button  on the toolbar,
- ▶ position the five objects underneath the factory information area (see Figure 11.42), and
- ▶ press the left-mouse button.

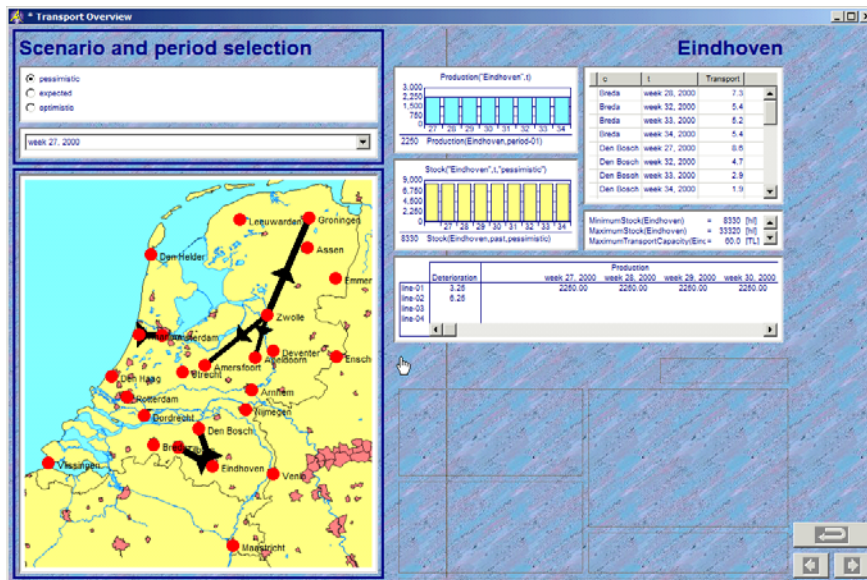


Figure 11.42: The copy and paste process illustrated

By now you should have enough experience to make a series of modifications to transform the factory data block into a distribution center data block. First add the following declaration at the end of the Transport Overview Declarations section.

Making the required modifications

```
STRING PARAMETER:
  identifier : CenterDescription
  definition : FormatString( "%e", DisplayedCenter )
```

The following list of actions now needs to be executed, using the detailed knowledge gained so far:

- ▶ change the string parameter `FactoryDescription` to the string parameter `CenterDescription` using the **Text** tab of the **Text Properties** dialog box of the copy of the text object,
- ▶ remove `MaximumTransportCapacity(DisplayedFactory)` from the **Contents** tab of the scalar object,
- ▶ find `Production(DisplayedFactory,t)` on the **Contents** tab of the production bar chart,
- ▶ change this to `Demand(DisplayedCenter,t,DisplayedScenario)`,
- ▶ find `Transport(DisplayedFactory,c,DisplayedPeriod,DisplayedScenario)` on the **Contents** tab of the factory transport composite table,

- ▶ change this to `Transport(f,DisplayedCenter,DisplayedPeriod,DisplayedScenario)`,
- ▶ open, in sequence, the **Contents** tab of the **Properties** dialog box associated with the table, the scalar object and the two bar charts, and
- ▶ replace all references to the element parameter `DisplayedFactory` with one to the element parameter `DisplayedCenter`.

11.3.11 Completing the page

At this point you should copy the three execution buttons (**Run Next**, **Run All** and **Restart**) from the *Production Overview* page, and paste them at the same position on the *Transport Overview* page. You could introduce a new template page for this purpose.

Copying the execution buttons

Finally, you could enhance the page by adding rectangles, changing text color and sizes as discussed in Subsection 11.2.8. Figure 11.43 will serve as a guide while completing the *Transport Overview* page on your screen.

Beautifying the page

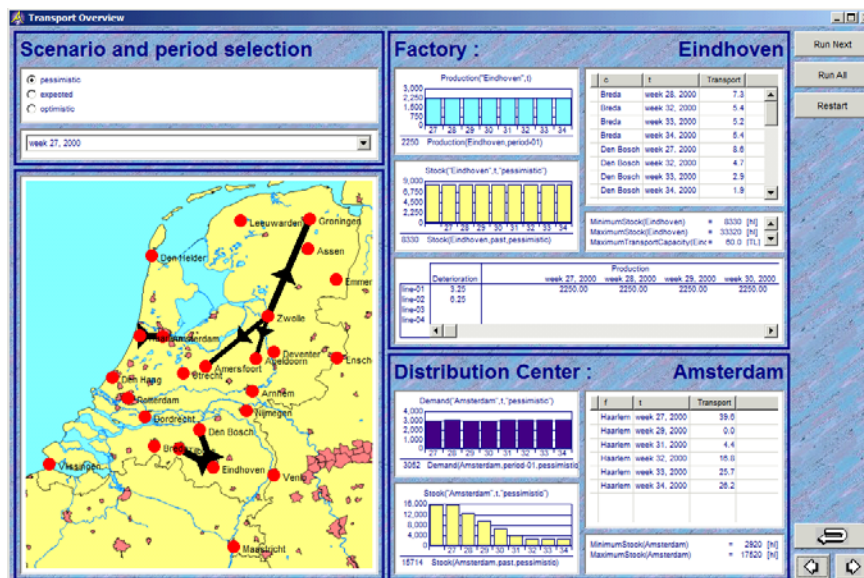


Figure 11.43: The completed *Transport Overview* page