USE AND EVALUATION

OF ELECTRE TRI 2.0a

Cristina Gómez, Jesús Ladevesa, Purushotham Muniganti

Karina Gibert, Aïda Valls

Juliol 2007

DEIM-RT-07-002

Index

Introduction

The Electre Tri 2.0a software is a multiple criteria decision aiding tool, designed to deal with sorting problems. The software is based on the Electre Tri method which has been developed by LAMSADE (Paris-Dauphine University, Paris, France). The software itself was developed through a collaboration of two research teams: LAMSADE and the Institute of Computing Science (Poznan University of technology, Poznan, Poland).

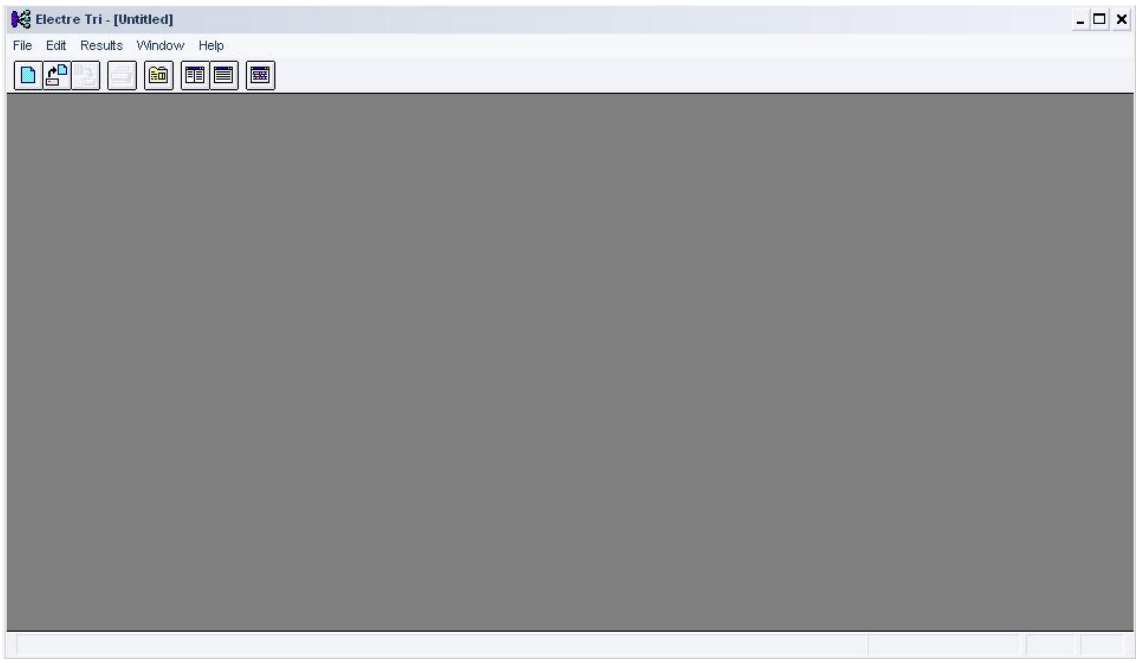
The Electre Tri can be downloaded from:

http://www.lamsade.dauphine.fr/english/software.html#TRI

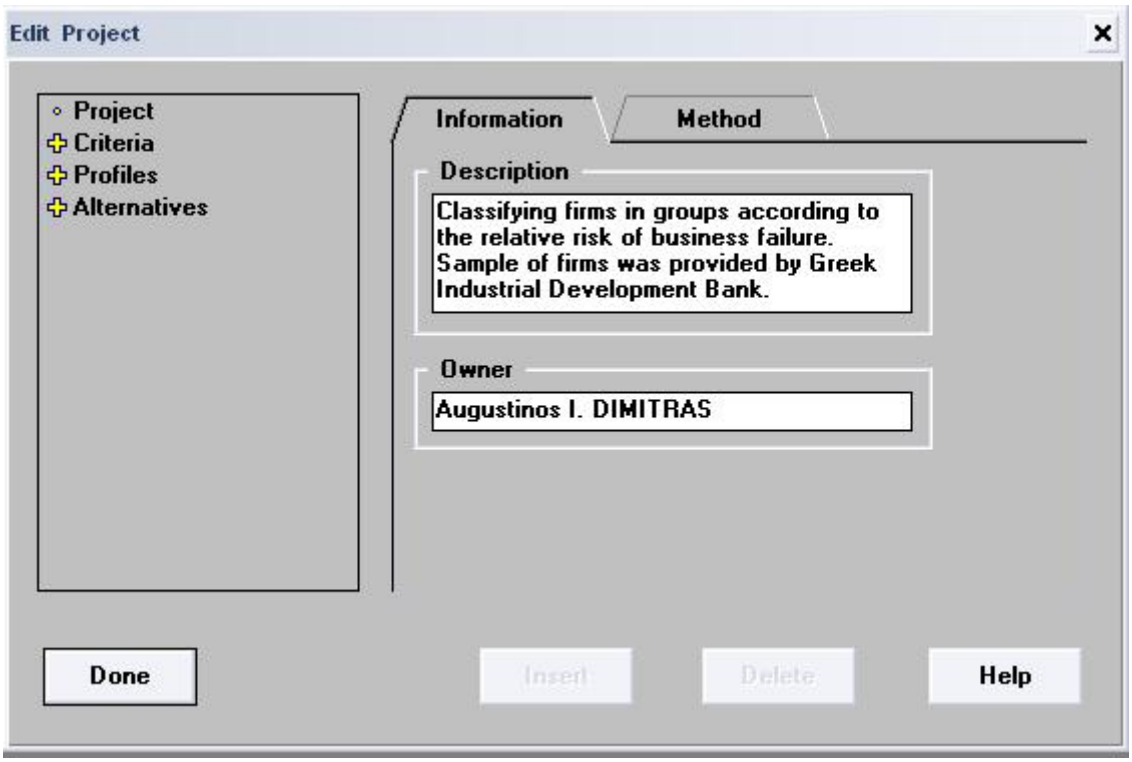
Users can execute it without need of installation by unzipping the zip file and opening the exe file. In this demo version users should input manually all the information because the load and save functions are disabled.

Editing

When you start Electre Tri program, you will see a window like this:



Users can create a new project by choosing in the File menu the option “New Project”, or by clicking on the first button of the General Menu toolbar. Then, a window with the configuration of all the characteristics of the project will appear.



On the left hand side of the window you'll find the different parts that can be edited. On the right hand side, the different items that can be edited for each part are shown.

In order to sort the data, Electre Tri uses three kinds of input data: **alternatives**, **criteria** and **profiles**.

The **alternatives** are the actual data you want to evaluate. They can either be inputted one by one, or by means of a text file. Possible alternatives might be different companies you want to order according to their achievements on different criteria.

The **criteria** indicate on what you want to evaluate the alternatives. Not every criterion might be as important as the other ones. In such a case, you can attribute weights to the different criteria. Examples of criteria are: price, quality, service,etc.

The **profiles** make up the boundaries between the different categories. The alternatives are divided over these different categories. There has to be set a profile for every criterion. There are three possible thresholds that can be set for every criterion.

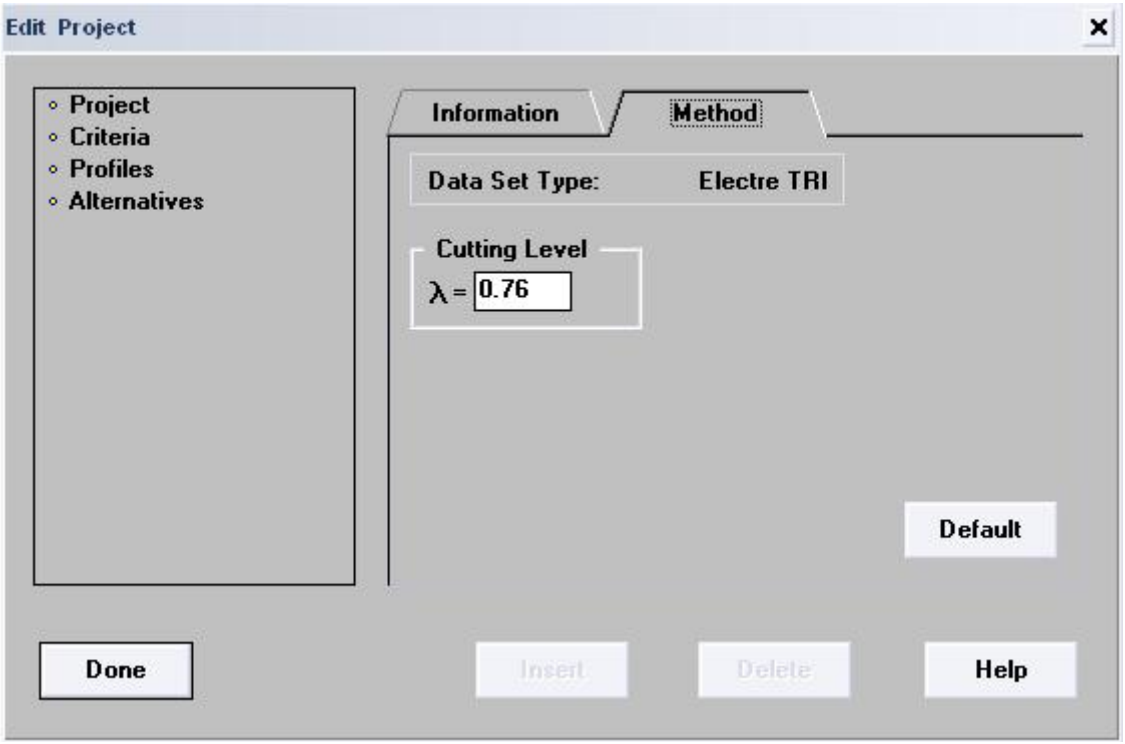
PROYECT

There are two tabs:

- **Information**: We can write a brief description of the project and the name of its developers.
- **Method**: The λ-cutting (λ should be in the range 0.5 and 1.0) level indicates how many of the criteria have to be fulfilled in order to assign an alternative to a specific category. λ is considered as the smallest value of the credibility index compatible with the assertion "a outranks bh".

  Example: If the cutting level is for instance 0.75, ¾ of the alternatives will have to respond to the rules making up that category.

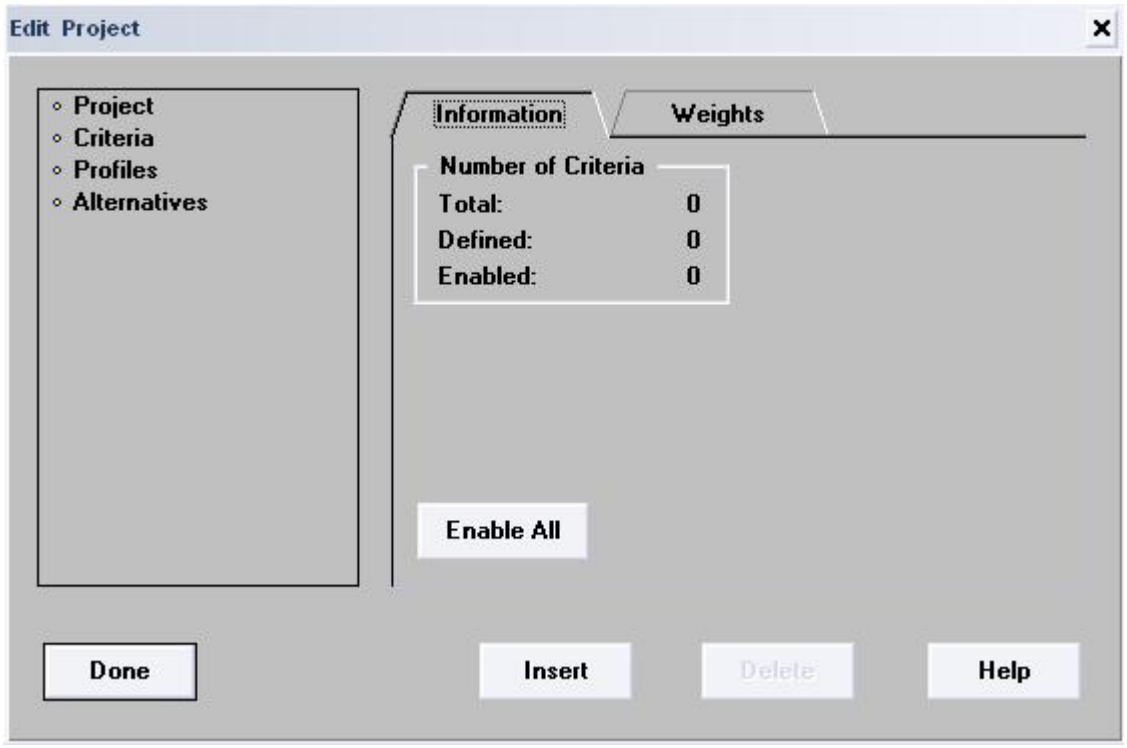
  The default value of the λ -cutting level is set at 0.76.



Electre Tri will make sure that each criterion, profile or alternative will automatically get a code. You can change these codes, but you have to make sure that you enter a unique value.

CRITERIA

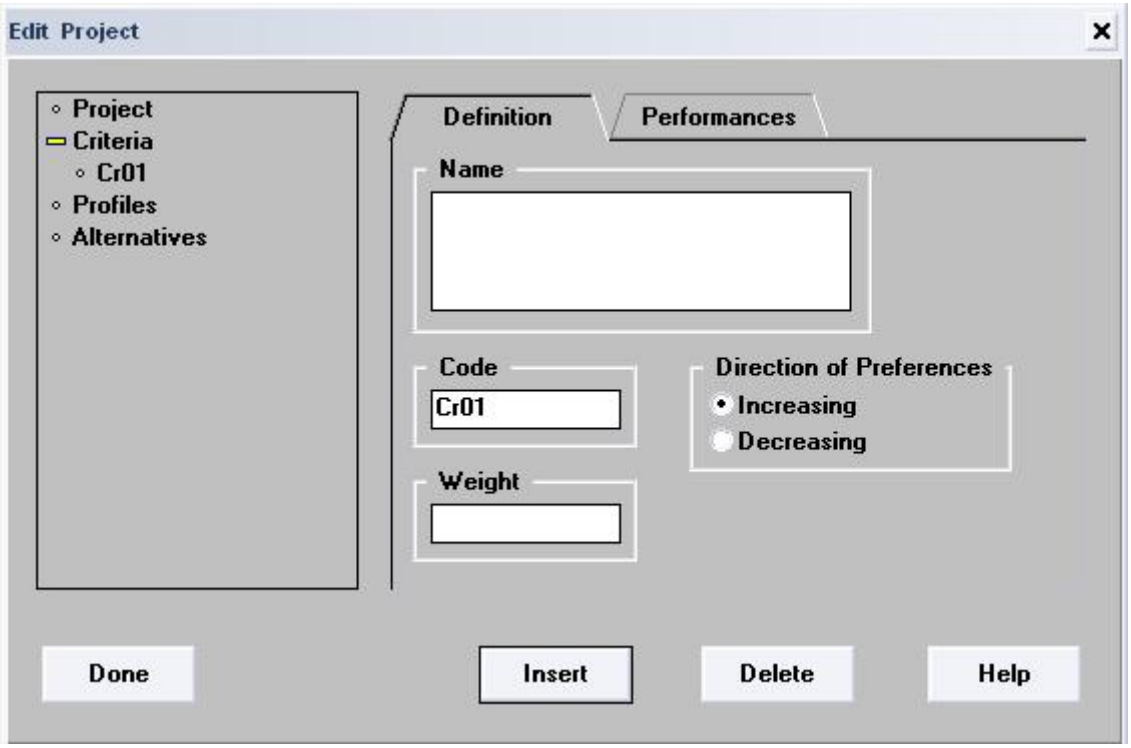
The criteria are the properties of the alternatives.



There are two tabs:

- **Information**: It resumes the number of criteria that the user has added.
- **Weights**: It shows the importance (weights) of all the criteria.

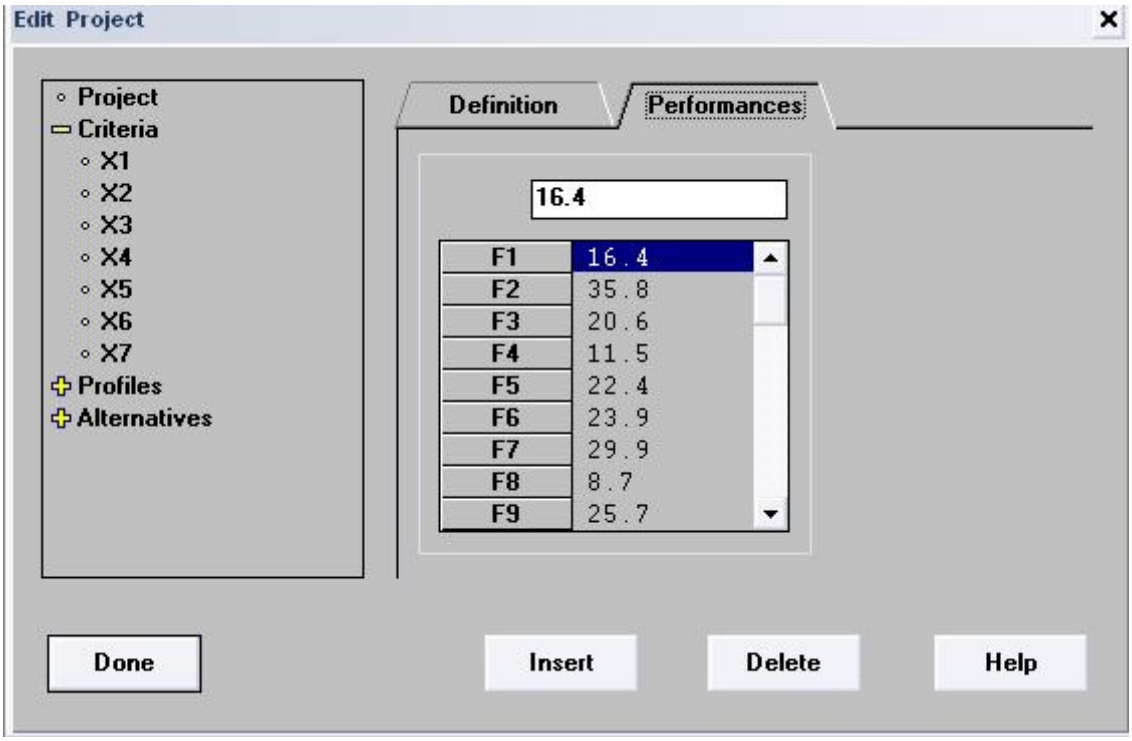
When the user inserts a new criterion he will be able to configure its properties, and its values will be reflected on the information and weights tab.



If you set the alternatives in an increasing way, this means that you'll think of the biggest values for the alternatives as the best ones. If you choose for decreasing, it will be the other way around.

The weight of a criterion indicates the importance of this criterion in relation to other ones.

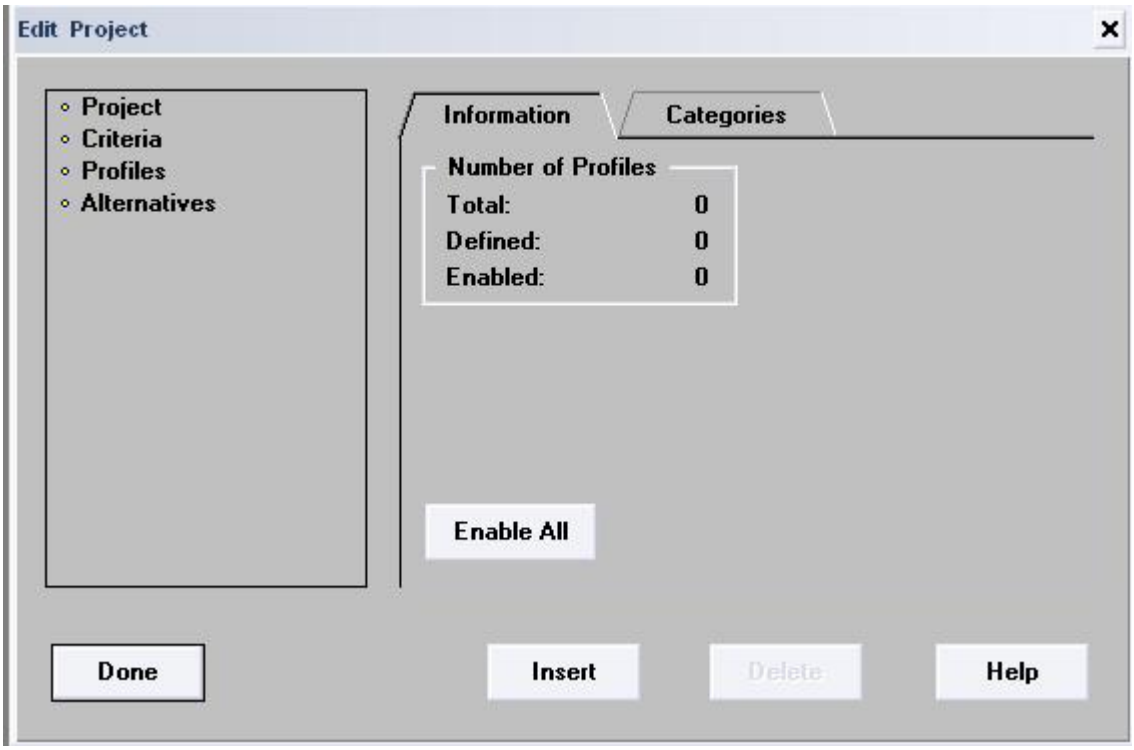
Any criterion could be deleted if the user wishes by clicking on the *"Delete" button*. In the same way, any number of criteria could be inserted by clicking on the *"Insert" button*.



The performances tab shows the values of this criterion for all the alternatives, and it could be modified for each alternative.

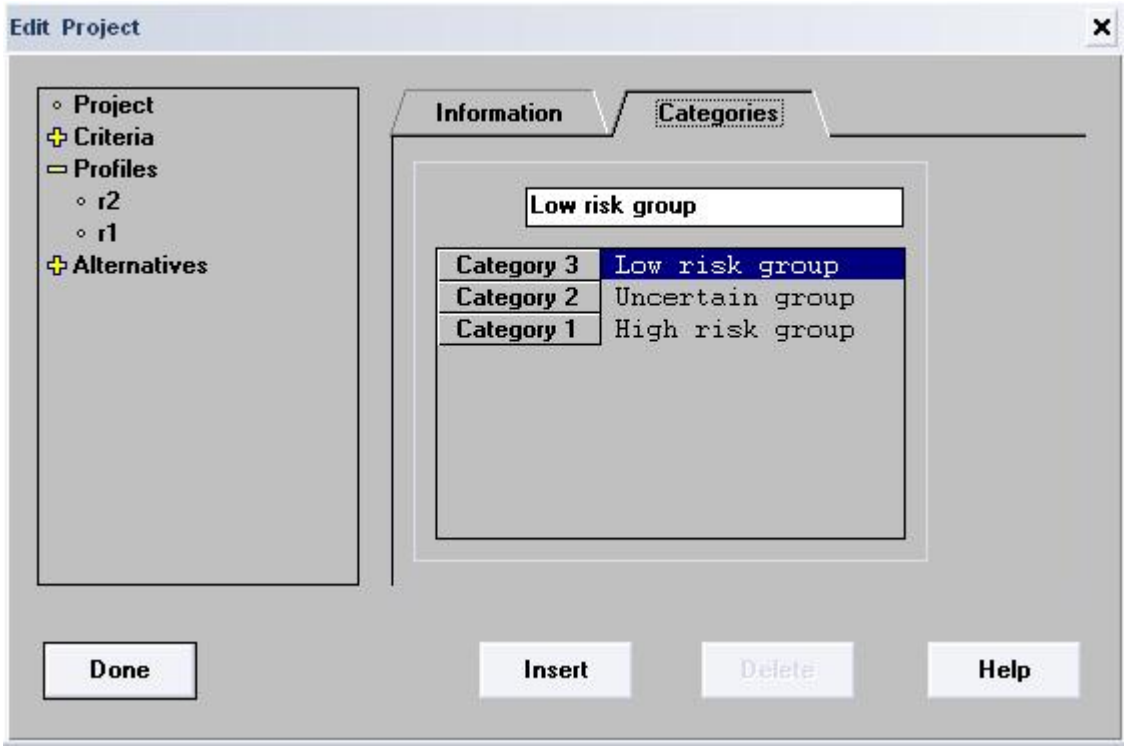
PROFILES

The profiles define the boundaries between the different categories.

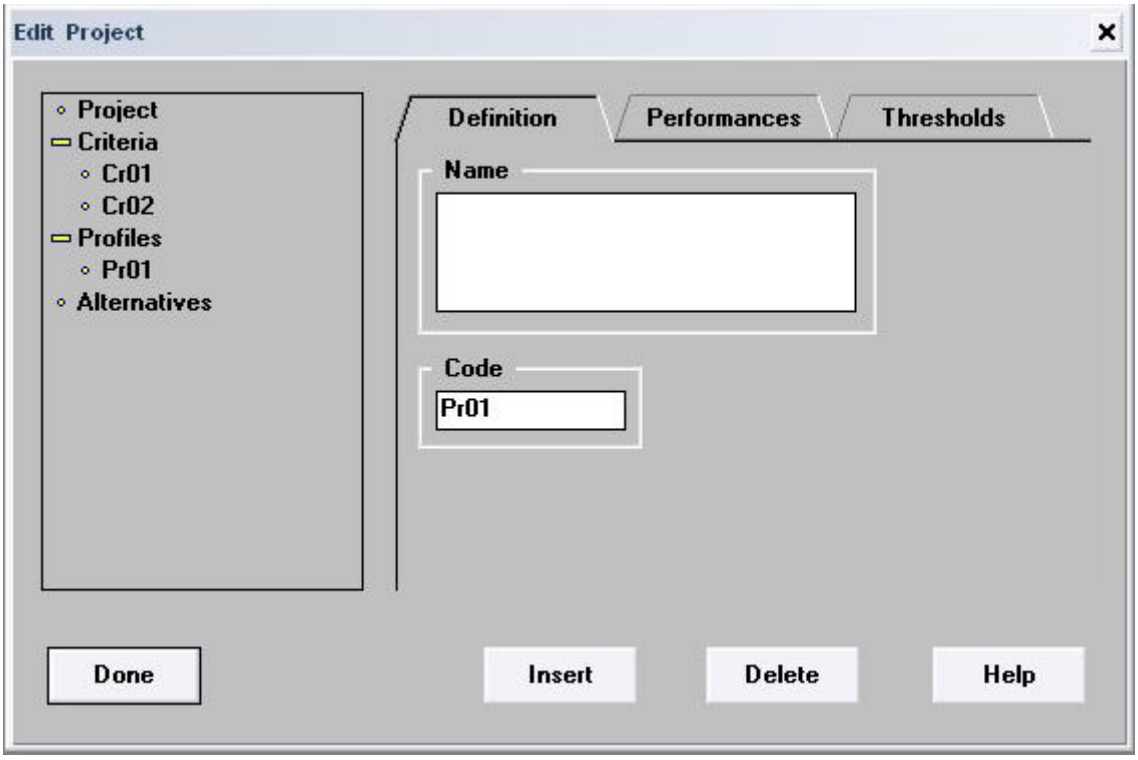


There are two tabs:

- **Information**: It shows a resume of all existing profiles.
- **Categories**: The tab Categories gives access to the different categories, if there are any. You can also change the name of the categories here. These names will appear if you show the assignment. The first name in the list has to be the name of the best category, the last name has to be the one of the worst category.

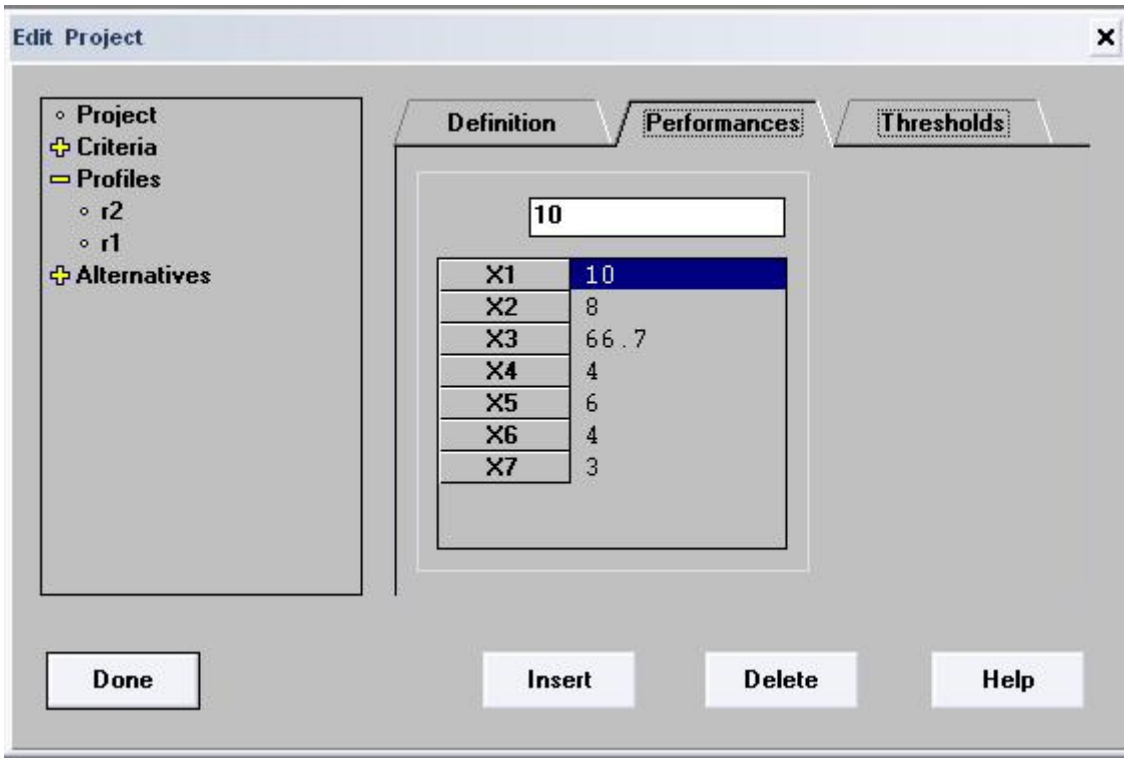


To insert a new profile, click on the *"Insert" button*. An empty profile will appear. The profiles must be always entered from the best to the worst.



There are three tabs:

- **Definition**: It shows the name and code of the profile.
- **Performances**: It allows modifying the values of the boundaries for each criterion.

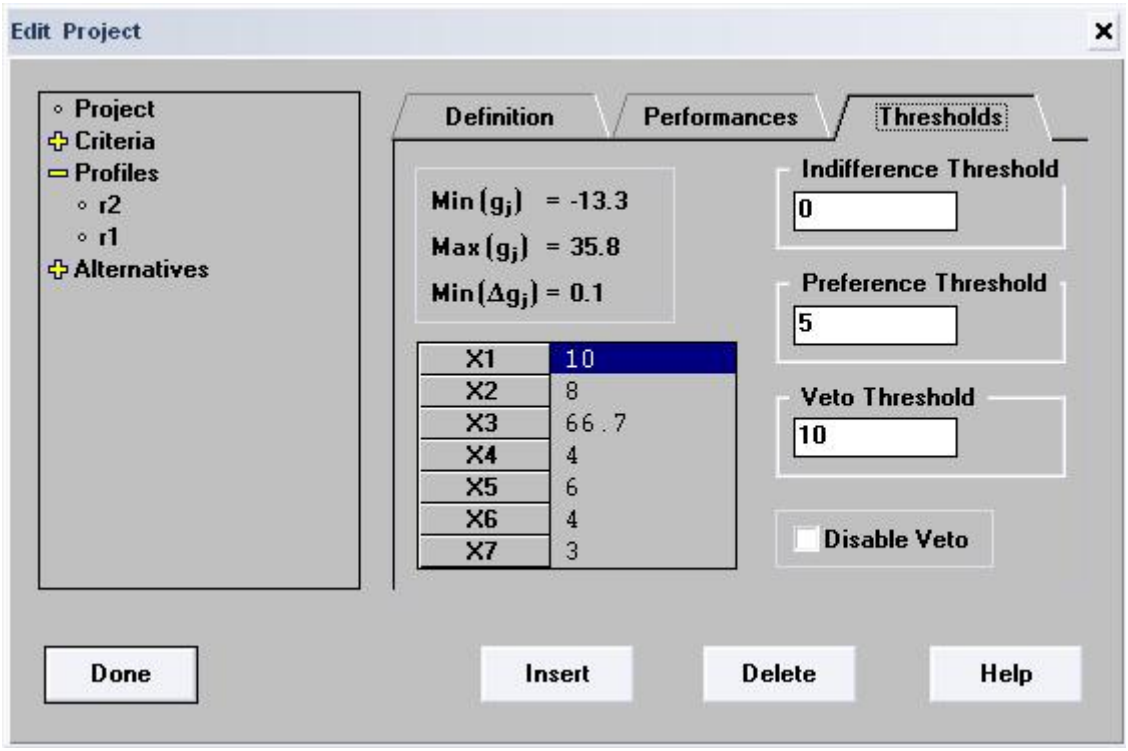


- **Thresholds**: There are three thresholds:

    1. **Indifference Threshold (I)**: The indifference threshold makes it possible to treat a certain value as a value that is closer to the value indicated by the profile. The number inputted shows the value that an alternative may differ from the value of the profile in order to be treated equally.

    2. **Preference Threshold (<,>)**: If a certain value reaches the preference threshold, it will be strongly preferred above the other values. The preference threshold indicates the difference between the value of the profile and the most preferred value.

    3. **Veto Threshold (R)**: The veto threshold indicates which value has to be reached to make sure that the alternative will make part of a certain category. As it is a veto value, it is enough that only the value indicated by this threshold is reached in order to sort the alternative in a specific category, no matter what the values of the other criteria of that alternative are.

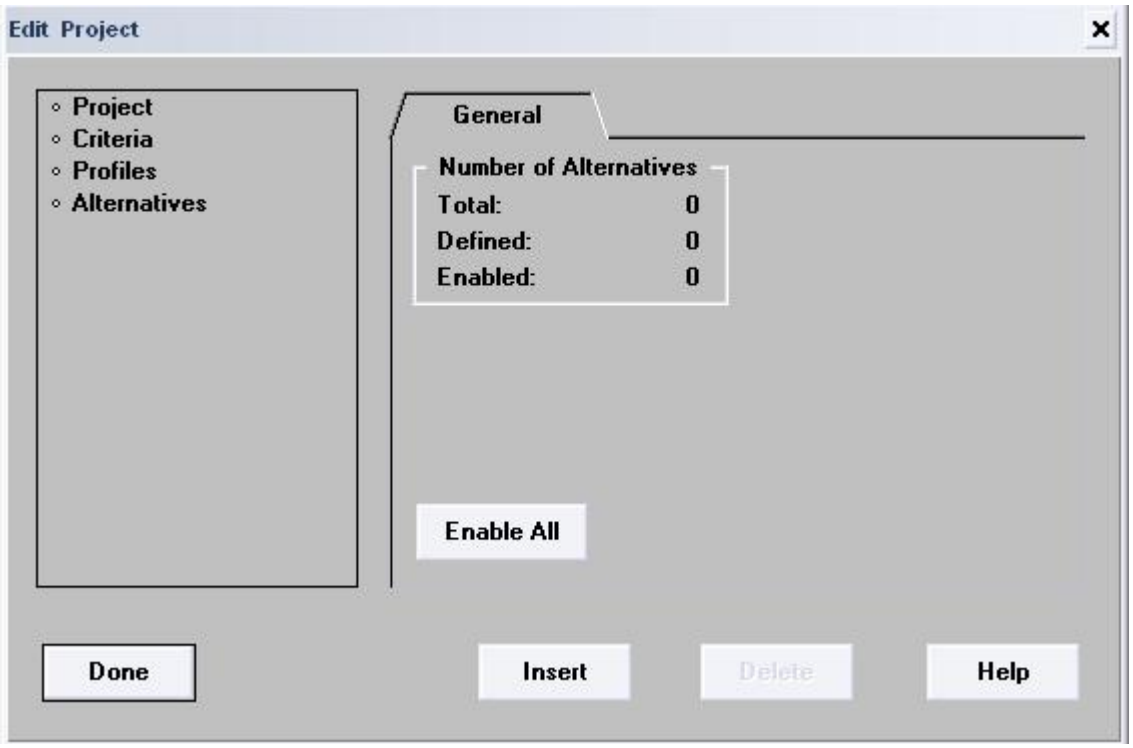


If you have entered alternatives into the project, there will be three different criterion values for each alternative. You find these values on the left top of the Thresholds tab.

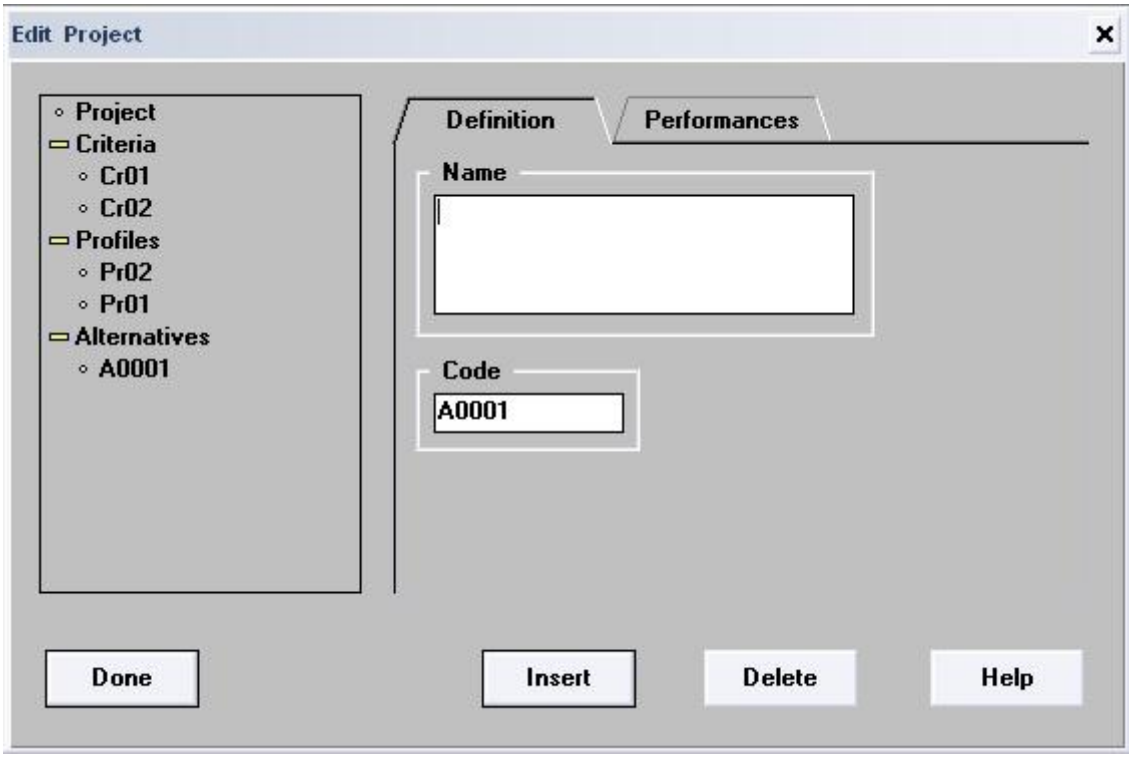
- **Min (gj)**: it indicates the smallest value that has been entered for the selected criterion.
- **Max (gj)**: it indicates the biggest value that has been entered for the selected criterion.
- **Min (Δgj)**: it indicates the smallest difference in values for the selected criterion.

ALTERNATIVES

The *"General" tab* specifies the total number of alternatives, the number of defined alternatives (those completely defined) and the number of enabled alternatives. The *"Enable all" button* gives the possibility to make all alternatives active.

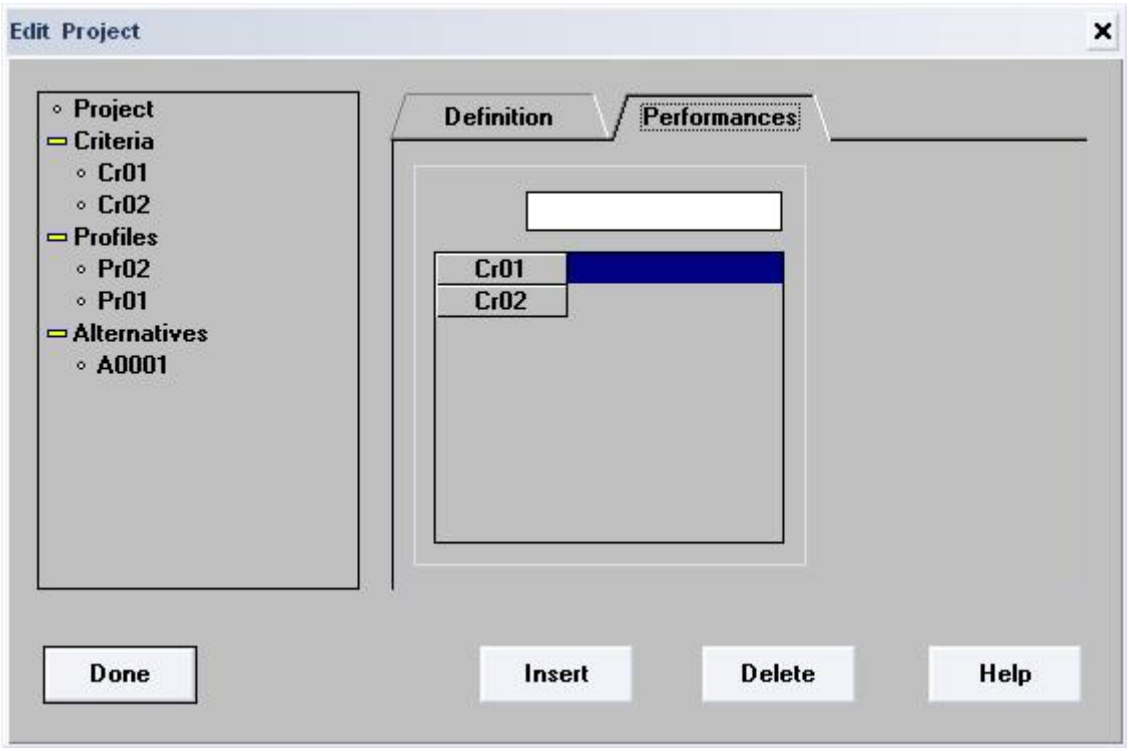


To insert a new and empty alternative just click on the *"Insert" button*, and a window like the one bellow will appear.



There are two tabs:

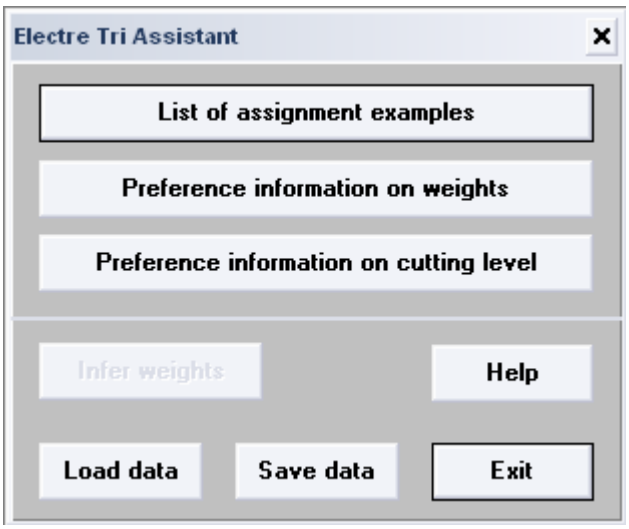
- **Definition**: The name and code of the alternative have to be specified here.
- **Performances**: The value of the alternative for all the criteria has to be specified here.



Electre Tri Assistant

This command provides support to the user in the definition of the values of parameters of the assignment model. The ELECTRE TRI Assistant functionalities are:

1. **"List of assignment examples"**: It inputs a list of alternative examples which belong to specific category/categories.
2. **"Preference information on weights"**: It gives eventually preferential information on the weights.
3. **"Preference information on cutting level"**: It sets the possible range for the cutting level.



Electre TRI Method

ELECTRE TRI assigns alternatives to predefined categories. The assignment of an alternative "**a**" results from the comparison with the profiles defining the limits of the categories.

The steps to compare an alternative (**a**) with a concrete profile (**b**) are:

1. Computation of partial concordance indices $\mathbf{c_j(b,a)}$ and $\mathbf{c_j(a,b)}$.

   Partial concordance index $c_j(a,b)$ ($c_j(b,a)$, respectively) expresses to which extend the statement **"a is at least as good as b (b is at least as good as a, respectively) considering criterion g_j"**.

   When g_j has an increasing direction of preference, index c_j (a,b) and c_j(b,a) are computed as follows (p and q are the preference and indiffirence thresholds respectively):

   If $g_j(a)<=g_j(b) - p_j(b)$, then $c_j(a,b)=0$
   If $(g_j(b)-p_j(b))<g_j(a)<=(g_j(b)-q_j(b))$,
   Then $c_j(a,b)=(g_j(a)-g_j(b)+p_j(b))/(p_j(b)-q_j(b))$
   If $(g_j(b)-q_j(b))<g_j(a)$, then $c_j(a,b)=1$

   If $g_j(a)>=(g_j(b)+p_j(b))$, then $c_j(b,a)=0$
   If $(g_j(b)+q_j(b))<=g_j(a)<(g_j(b)+p_j(b))$,
   Then $c_j(b,a)=(g_j(b)-g_j(a)+p_j(b))/(p_j(b)-q_j(b))$
   If $g_j(a)<(g_j(b)+q_j(b))$, then $c_j(b,a)=1$

   When g_j has a decreasing direction of preference, index c_j (a,b) and cc_j (b,a) are computed as follows:

   If $g_j(a)>=(g_j(b)+p_j(b))$, then $c_j(a,b)=0$
   If $(g_j(b)+q_j(b))<=g_j(a)<=(g_j(b)+p_j(b))$,
   Then $c_j(a,b)=(g_j(b)-g_j(a)+p_j(b))/(p_j(b)-q_j(b))$

If $g_j(b)+q_j(b)>g_j(a)$, then $c_j(a,b)=1$

If $g_j(a)<=(g_j(b)-p_j(b))$, then $c_j(b,a)=0$
If $(g_j(b)-p_j(b))<=g_j(a)<g_j(b)-q_j(b)$,
Then $c_j(b,a)=((g_j(a)-g_j(b)+p_j(b))/(p_j(b)-q_j(b))$
if $g_j(a)>(g_j(b)-q_j(b))$, then $c_j(b,a)=1$

2. Computation of the global concordance indices **c(b,a)** and **c(a,b)**.

Global concordance indices c(b,a) (c(a,b), respectively) express to which extend the evaluations of a and b on all criteria are concordant with the assertion **"a outranks b (b outranks a, respectively)"**.c(b,a) and c(a,b) are computed as follows:

$$c(a,b)=\frac{\sum_{j\in F} k_j\ c_j(a,b)}{\sum_{j\in F} k_j}$$

$$c(b,a)=\frac{\sum_{j\in F} k_j\ c_j(b,a)}{\sum_{j\in F} k_j}$$

3. Computation of the partial discordance indices **d_j(a,b)** and **d_j(b,a)**.

Partial discordance index $d_j(a,b)$ ($d_j(b,a)$, respectively) expresses to which extend the criterion g_j is opposed to the assertion "a is at least as good as b". A criterion g_j is said to be discordant with to assertion **"a outranks b is on this criterion b is preferred to a (b Pa, i.e., $c_j(b,a)=1$ and $c_j(a,b)=0$)"**. In the case of increasing preferences, the criterion g_j opposes a veto when the difference $g_j(b)-g_j(a)$ exceeds the veto threshold $v_j(b)$.

When g_j has an increasing direction of preference, index $d_j(a,b)$ and $d_j(b,a)$ are computed as follows (p and q are the preference and indiffirence thresholds respectively):

If $g_j(a)>(g_j(b)-p_j(b))$ then $d_j(a,b)=0$
If $(g_j(b)-v_j(b)<g_j(a))<=(g_j(b)-p_j(b))$
Then $d_j(a,b)=(g_j(b)-g_j(a)-p_j(b))/(v_j(b)-p_j(b))$

If $(g_j(b)-v_j(b)) >= g_j(a)$ then $d_j(a,b)= 1$

If $g_j(a) <= (g_j(b) + p_j(b))$ then $d_j(b,a)= 0$

If $(g_j(b) + p_j(b)) < g_j(a) <= (g_j(b) + v_j(b))$

Then $d_j(b,a) = (g_j(a)-g_j(b)-p_j(b))/(v_j(b)-p_j(b))$

If $g_j(a) > (g_j(b) + v_j(b))$ then $d_j(b,a)=1$

When g_j has a decreasing direction of preference, index $d_j(a,b)$ and $d_j(b,a)$ are computed as follows:

If $g_j(a) <= (g_j(b) + p_j(b))$ then $d_j(a,b)= 0$

If $(g_j(b) + p_j(b)) < g_j(a) <= (g_j(b) + v_j(b))$,

Then $d_j(a,b)= (g_j(a)-g_j(b)-p_j(b))/(v_j(b)-p_j(b))$

If $(g_j(b)+v_j(b)) < g_j(a)$, then $d_j(a,b) = 1$

If $g_j(a) > (g_j(b)-p_j(b))$, then $d_j(b,a) = 0$

If $g_j(b) - v_j(b) < g_j(a) <= (g_j(b)-p_j(b))$,

Then $d_j(b,a) = (g_j(b)-g_j(a)-p_j(b))/(v_j(b)-p_j(b))$

If $g_j(a) <= (g_j(b) - v_j(b))$, then $d_j(b,a)= 1$

4. Computation of the credibility indices **σ(a,b)** and **σ(b,a)**.

The degree of credibility of the outranking relation σ(a,b) (σ(b,a), respectively) expresses to which extend **"a outranks b (b outranks a, respectively) according to the global concordance index c(a,b) and to the discordance indices $d_j(a,b)$"**.

The computation of the credibility index σ(a,b) is grounded on the following principles:

1) When no criteria are discordant, the credibility of the outranking relation σ(a,b) is equal to the concordance index σ(a,b),

2) When a discordant criterion opposes a veto to the assertion "a outranks b"((i.e., $d_j(a,b)=1$), then credibility index σ(a,b) becomes null (the assertion "a outranks b" is not credible at all),

3) When a discordant criterion is such that $c(a,b)<d_j(a,b)<1$, the credibility index σ(a,b) becomes lower than the concordance index c(a,b), due to the effect of the opposition on this criterion.

It results from these principles that the credibility index σ(a,b) corresponds to the concordance index c(a,b) weakened by eventual veto effects. More precisely, the value of σ(a,b) is computed as follows (σ(b,a) is computed similarly):

$$\sigma(a,b)=c(a,b)\prod_{j\in\overline{F}}\frac{1-d_j(a,b)}{1-c(a,b)} \text{ where } \overline{F}=\{j\in F / d_j(a,b)>c(a,b)\}$$

CONSISTENCY OF CATEGORIES

The ordered *p+1* categories $C_1, C_2, \ldots, C_{p+1}$ are defined in ELECTRE TRI by p profiles $b_1, b_2, \ldots, b_p$, b_h being the upper limit of category C_h and the lower limit of category C_{h+1}, $h=1, 2, \ldots, p$. The profiles should fulfill next conditions:

- Categories should be ordered (in other case, it is not possible to use the method).
- No alternative can be indifferent to more than one profile (a situation in which aIb_h and aIb_{h+1} would implicitly mean that the category delimited by the profiles b_h and b_{h+1} is "insufficiently wide"). It is a sufficient condition for the preceding property.

Results

ASSIGNMENT BY CATEGORY

To show the results of the project ordered by category, the user can click on the *"Results > Assignment by Category" button*. This window shows three columns. The first one has the different categories, while the middle and right ones shows the alternatives assigned to the categories selected by the pessimistic and optimistic assignment procedure.

Pessimistic assignment (or conjunctive) procedure:

An action a will be assigned to the highest category C_h such that aSb_{h-1}.

a) Compare a successively to b_r, for r=k-1,k-2, ..., 0,

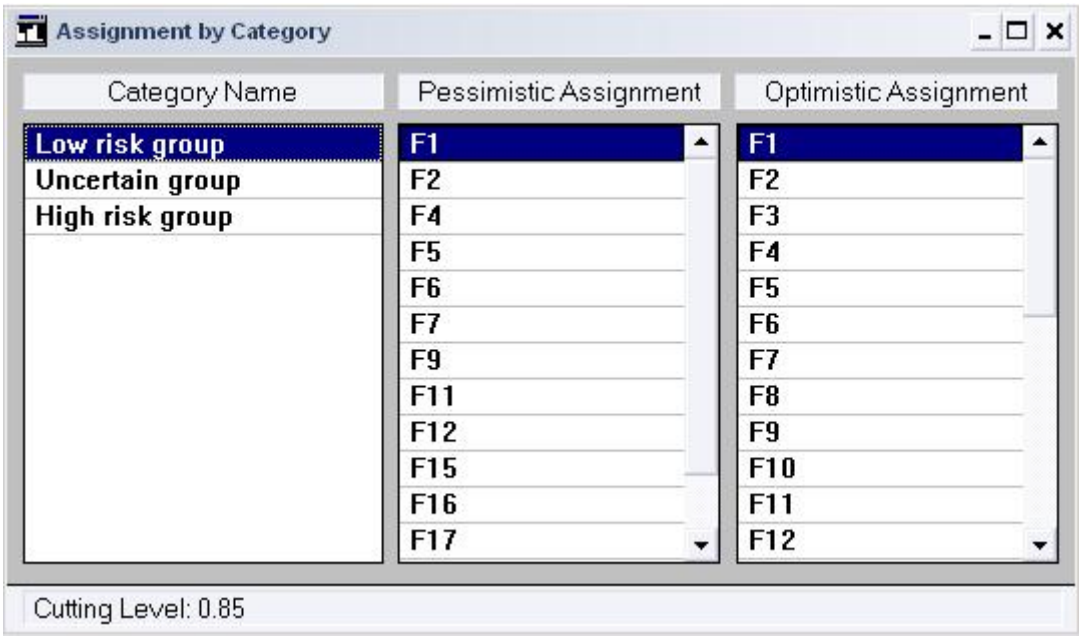
b) The limit b_h is the first profile encountered such that aSb_h, assign a to category C_{h+1}.

Optimistic assignment (or disjunctive) procedure:

An action a will be assigned to the lowest category C_h such that $b_h > a$.

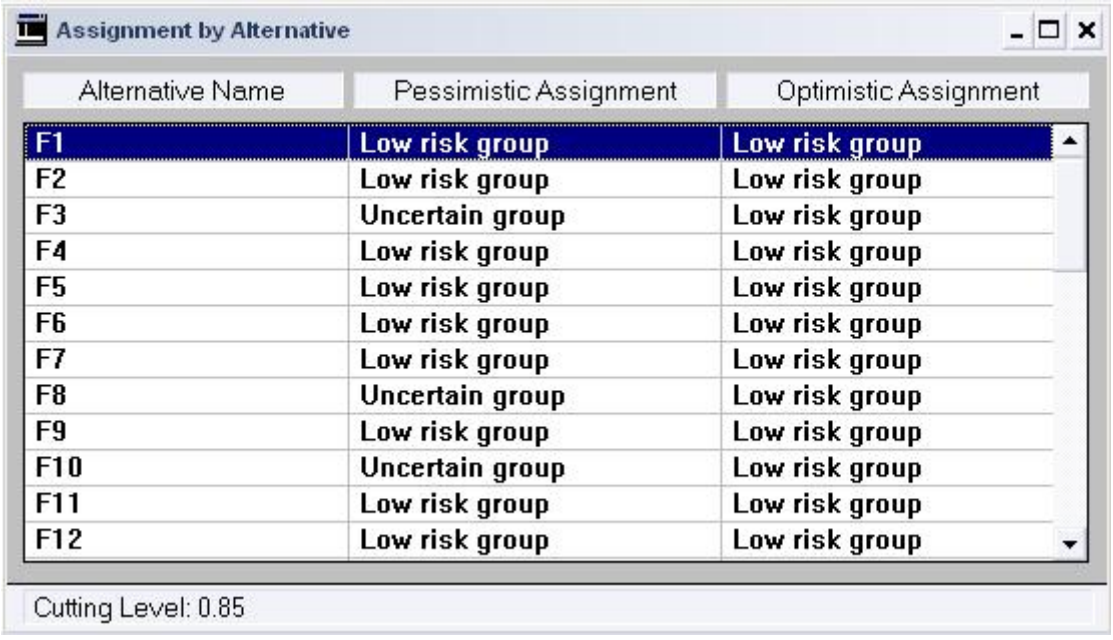
a) Compare a successively to b_r, i=1, 2, ..., k-1,

b) The limit b_h is the first encountered profile such that $b_h > a$, assign a to category C_h.



ASSIGNMENT BY ALTERNATIVE

To show the results of the project ordered by alternative, the user can click on the "Results > Assignment by alternative" button. Then, a Window like the one bellow will be opened.



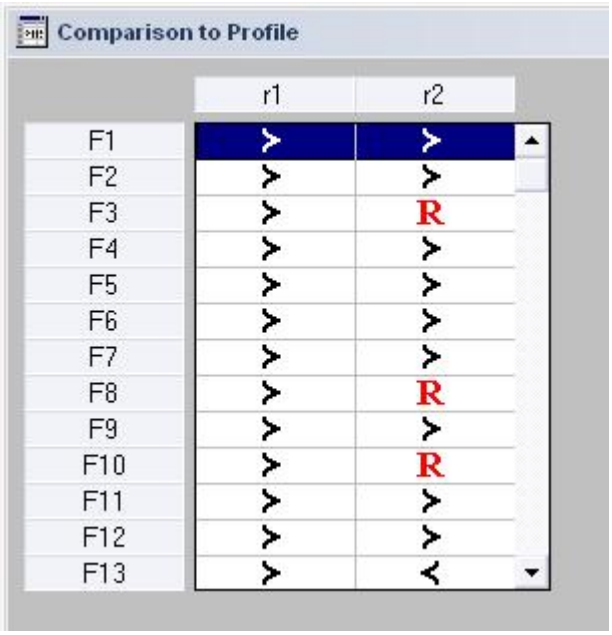
On the left column we find the alternative names, on the middle one the pessimistic assignments and on the right one the optimistic assignments.

COMPARISON TO PROFILE

This command displays the intermediary results concerning the comparisons of all alternatives to each limit profile.

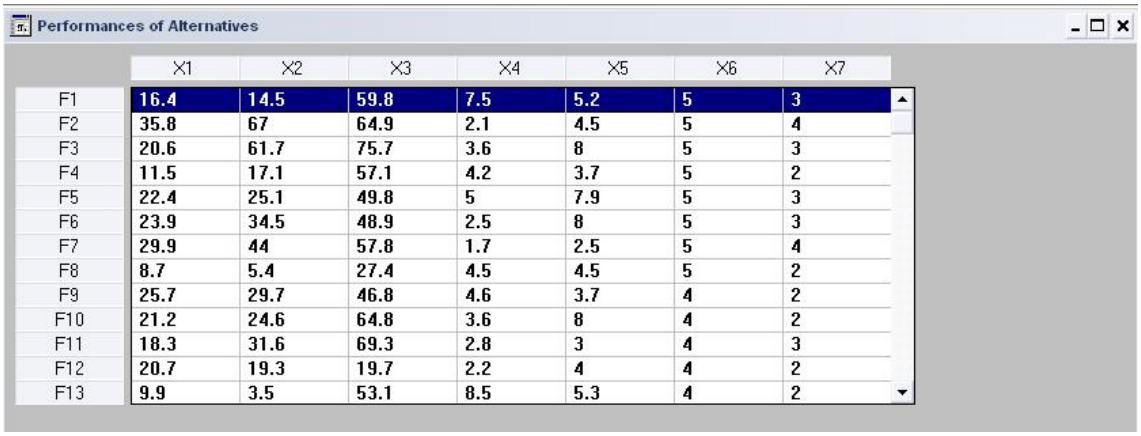
The symbols used are:

- < (or >) for a preference
- I for indifference.
- R for incomparability.



PERFORMANCES OF ALTERNATIVES

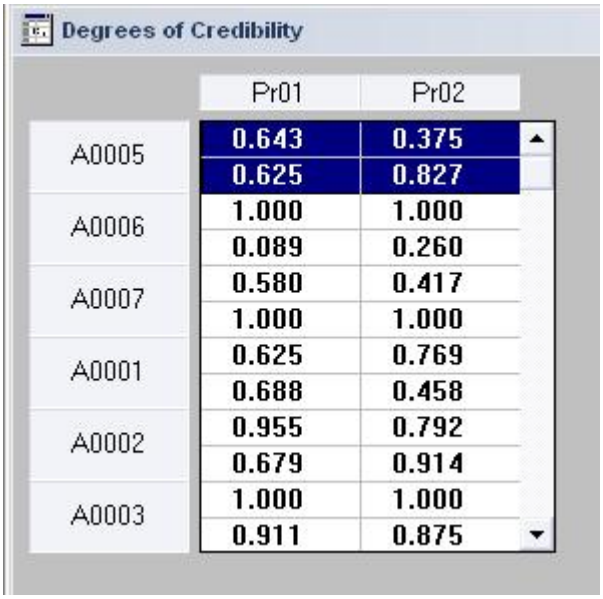
This command shows the performance matrix of the project. It resumes all the values that have been assigned to each alternative for each criterion.



	X1	X2	X3	X4	X5	X6	X7
F1	16.4	14.5	59.8	7.5	5.2	5	3
F2	35.8	67	64.9	2.1	4.5	5	4
F3	20.6	61.7	75.7	3.6	8	5	3
F4	11.5	17.1	57.1	4.2	3.7	5	2
F5	22.4	25.1	49.8	5	7.9	5	3
F6	23.9	34.5	48.9	2.5	8	5	3
F7	29.9	44	57.8	1.7	2.5	5	4
F8	8.7	5.4	27.4	4.5	4.5	5	2
F9	25.7	29.7	46.8	4.6	3.7	4	2
F10	21.2	24.6	64.8	3.6	8	4	2
F11	18.3	31.6	69.3	2.8	3	4	3
F12	20.7	19.3	19.7	2.2	4	4	2
F13	9.9	3.5	53.1	8.5	5.3	4	2

DEGREES OF CREDIBILITY

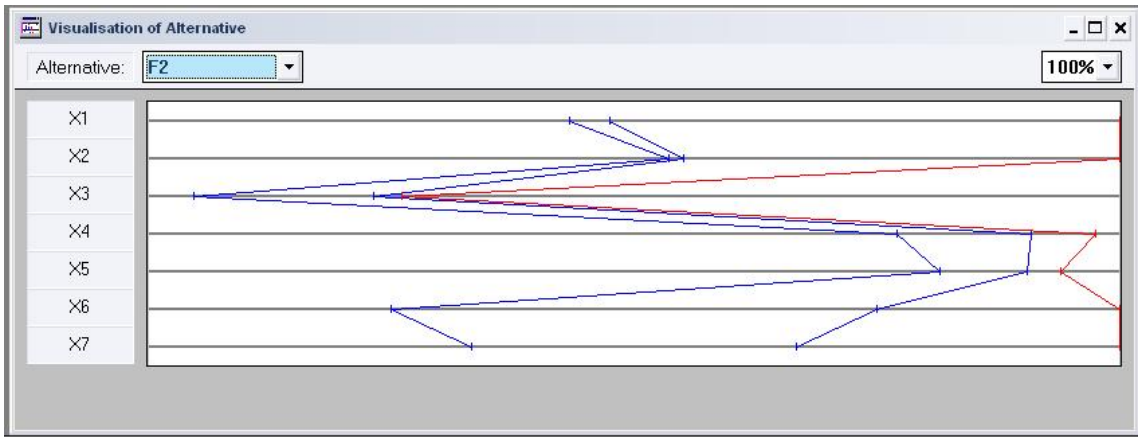
Display degree of credibility of the outranking relation between each profile bh and each alternative a. Each cell of the table contains two values s(a,bh) and s(bh,a). The value at the top of the cell corresponds to s(a,bh), the one at the bottom of the cell corresponds to s(bh,a).



	Pr01	Pr02
A0005	0.643	0.375
	0.625	0.827
A0006	1.000	1.000
	0.089	0.260
A0007	0.580	0.417
	1.000	1.000
A0001	0.625	0.769
	0.688	0.458
A0002	0.955	0.792
	0.679	0.914
A0003	1.000	1.000
	0.911	0.875

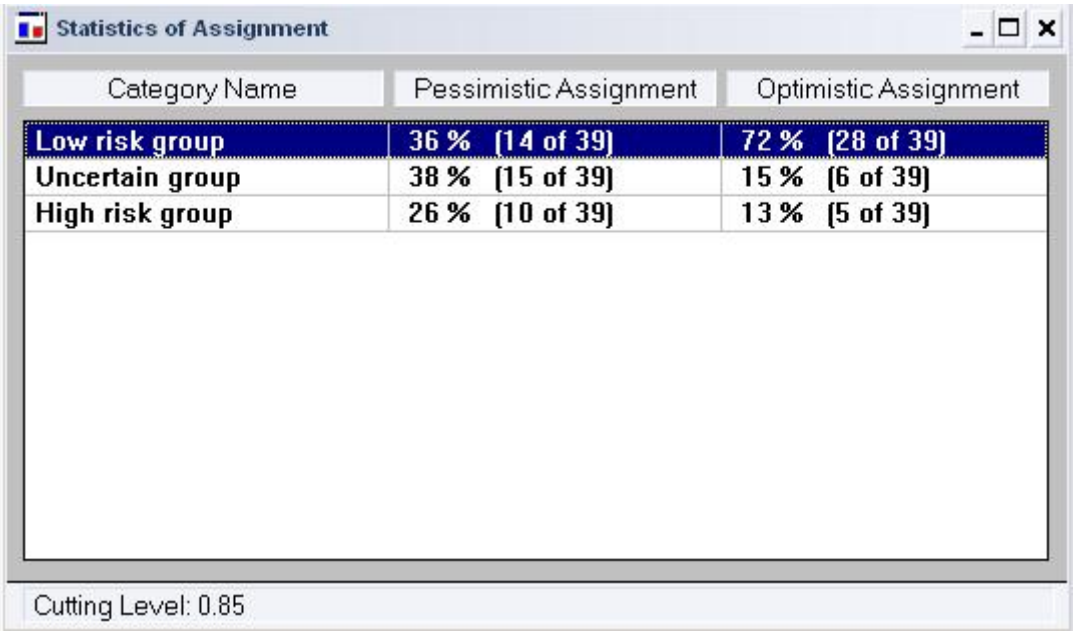
VISUALISATION OF ALTERNATIVE

This command provides a visual representation of an alternative and the profiles. The representation enables to visualize one alternative at a time (in red) together with the profiles specifying the limits of the categories (in blue).



STADISTICS OF ASSIGNMENT

This command gives the proportion of alternatives assigned to each category for both assignment rules (optimistic and pessimistic).



Presentation of Electre Tri (Window)

CASCADE

This command may be used to organize the different windows open on the desktop in such a way that their titles are always visible despite overlapping.

TILE

This command may be used to organize the different windows open on the desktop in such a way that they are all entirely visible.

ARRANGE ICONS

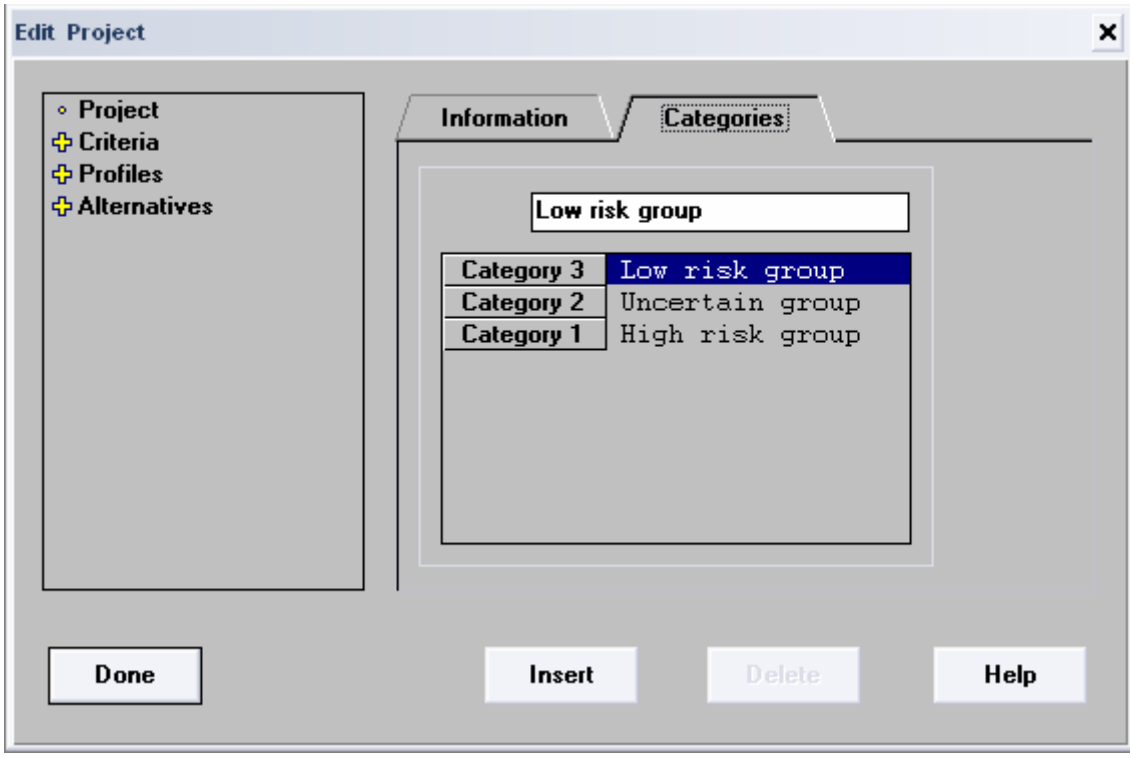
This command may be used to reorganize the different icons on the desktop, they will be gathered at the bottom of the screen.

CLOSE ALL

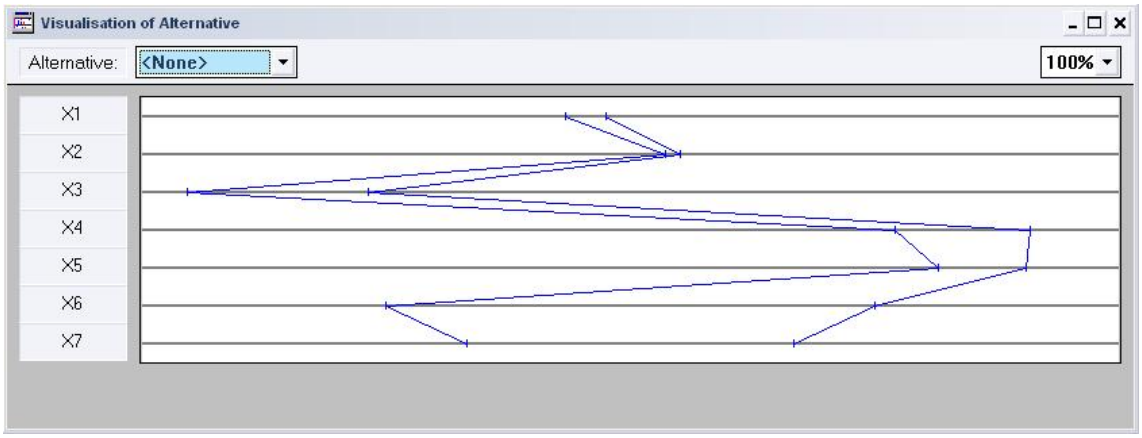
This command closes all windows and icons open on the desktop.

Example: risk of business failure

In this example Electre Tri classifies firms in groups according to the relative risk of business failure. Sample of firms was provided by Greek Industrial Development Bank. There are two specified profiles: one which represents the low risk firms and other which represents the high risk ones. With this information Electre Tri automatically creates three categories: high risk group, uncertain group and low risk group.



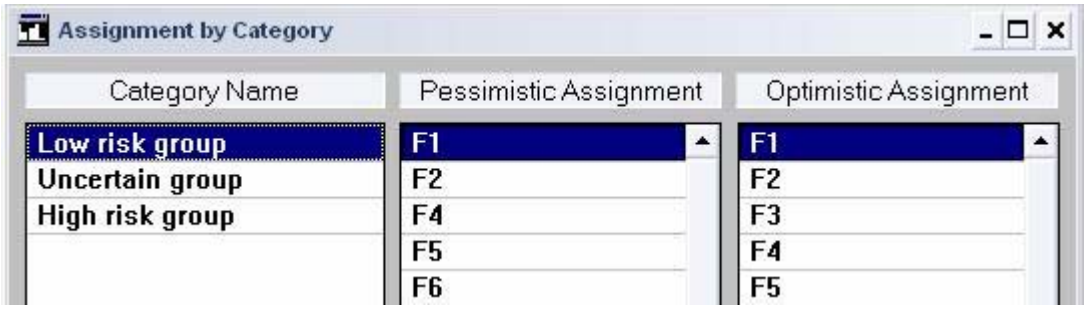
In the uncertain group Electre Tri puts all the alternatives that don't fit any of the profiles indicated. By the Visualization of Alternative command the limits of each category for each criterion are shown.



In this example the limits of each profile are very similar in many criteria, like the net income (X2) or the earning before interest and taxes (X1), criteria which are not so important for the risk of business failure; and are a lot different at the last criteria, like managers work experience (X6), criteria which really indicate the risk of business failure. The “space” between the profiles owns to the uncertain group.

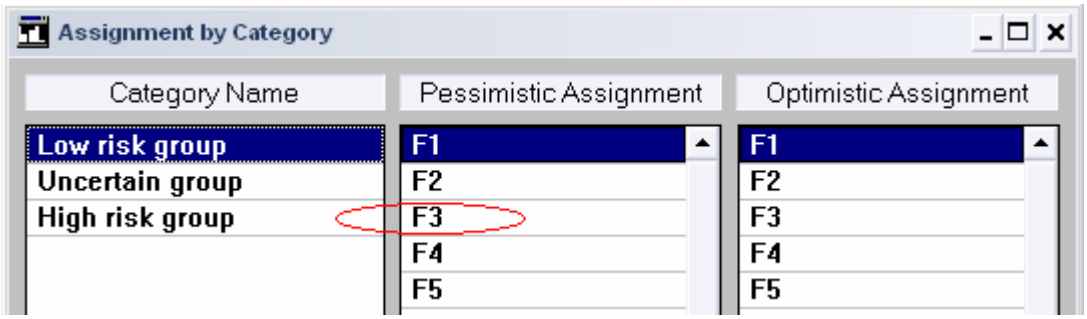
This is useful to the user because he can, just with a quick look, see if the values of the profiles indicated are right or they need to be modified because the profiles are way too similar, too different, etc.

By the Assignment by Category command Electre Tri shows how are all the alternatives assigned to each category, using the pessimistic and the optimistic assignment.



In this example the five best alternatives for the low risk group are F1, F2, F4, F5 and F6 according to the pessimistic assignment and F1, F2, F3, F4 and F5 according to the optimistic one. The results are the same for each algorithm except for F3, which is not assigned to this group by the pessimistic assignment because,

with the cutting level specified (0.85), the first group this alternative fits is the uncertain one. If we change the cutting level to a lower and less restrict value, like 0.65, Electre Tri will assign this alternative to this group.



Conclusions

Electre Tri is a software for taking multicriteria decisions. With it, a user can obtain the most adequate alternative according to his own preferences.

After the insertion of the selection criteria, the different alternatives and the profiles defined by the user, Electre Tri shows the user its results in a graphic way and acoording to different options.

During the accomplishment of the analysis of this software, the main dificulties we found were:

- Options blocked due to the use of a DEMO versión. Electre Tri (in its free versión) only allows the recuperation of projects already created, but it didn't allow saving neither the modifications done on them nor the creation of new projects. In the same way, in this version, the loading of alternatives from text files is not posible and they must be inputted manually.
- The hard comprehension of the method used for the comparation among alternatives and the classification of them in categories. This fact increased dramatically the understanding of its functionality. Initially, it results very complex and little accesible. The assignation due to an optimistic or pessimistic aproach also was an impediment for this task.

On the other hand, Electre Tri simplifies the arduous task of the expert making easy the carrying out of the calculation of multicriteria decisions with a great amount of alternatives. The specification of categories, not available in previous versions of Electre Tri (as III and IV), it's an improvement of the software that favours its use.

References

- MOSSEAU, V.; SLOWINSKI, R.; ZIELNIEWICCZ, P. **"ELECTRE TRI 2.0a. Methodological Guide and User's Manual"**. Document Du Lamsade. Paris, France: Université Paris – Dauphine. 1999.
- FIGUEIRA, J.; GRECO, S.; EHRGOTT, M. **"MULTIPLE CRITERIA DECISION ANALYSIS: State of the Art Surveys"**. Springer Science + Business Media, Inc. 2005.
- http://www.lamsade.dauphine.fr/english/software.html#TRI (25/02/2007)

Appendixes

APPENDIX 1: BINARY RELATIONS

There are three binary relations:

- **Preference** (< or >):
  - $a > b_h \leftrightarrow aSb_h$ and not b_hSa
  - $a < b_h \leftrightarrow$ not aSb_h and b_hSa

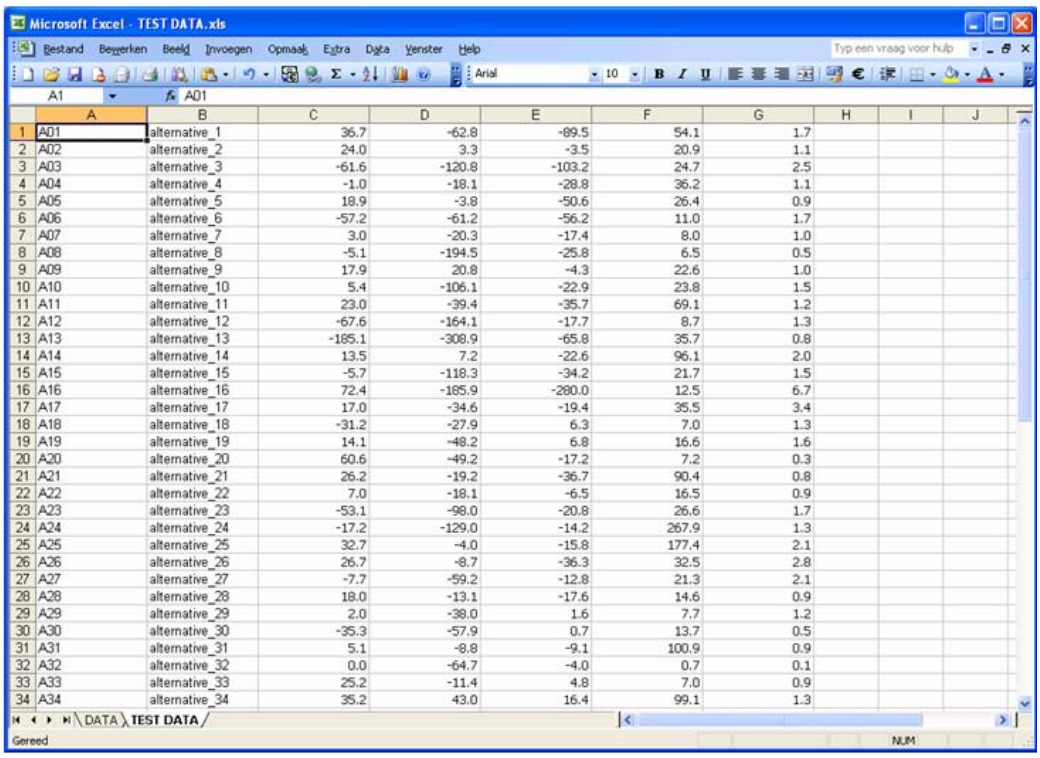
- **Indifference** (I):
  - $aIb_h \leftrightarrow aSb_h$ and b_hSa

- **Incomparability** (R):
  - $aRb_h \leftrightarrow$ not aSb_h and not b_hSa

APPENDIX 2: IMPORT ALTERNATIVES FROM AN EXCEL FILE

Existing data could be imported from an Excel file. Its structure has to be:

- **Rows**: Each row is equivalent to an alternative.
- **First column**: It contains the code of the alternative, no longer than 8 characters and without spaces.
- **Second column**: It contains the name of the alternative, no longer than 255 characters and without spaces.
- **The rest of the columns**: It contains the value of the alternative for all the criteria.



	A	B	C	D	E	F	G	H	I	J
1	A01	alternative_1	36.7	-62.8	-89.5	54.1	1.7			
2	A02	alternative_2	24.0	3.3	-3.5	20.9	1.1			
3	A03	alternative_3	-61.6	-120.8	-103.2	24.7	2.5			
4	A04	alternative_4	-1.0	-18.1	-28.8	36.2	1.1			
5	A05	alternative_5	18.9	-3.8	-50.6	26.4	0.9			
6	A06	alternative_6	-57.2	-61.2	-56.2	11.0	1.7			
7	A07	alternative_7	3.0	-20.3	-17.4	8.0	1.0			
8	A08	alternative_8	-5.1	-194.5	-25.8	6.5	0.5			
9	A09	alternative_9	17.9	20.8	-4.3	22.6	1.0			
10	A10	alternative_10	5.4	-106.1	-22.9	23.8	1.5			
11	A11	alternative_11	23.0	-39.4	-35.7	69.1	1.2			
12	A12	alternative_12	-67.6	-164.1	-17.7	8.7	1.3			
13	A13	alternative_13	-185.1	-308.9	-65.8	35.7	0.8			
14	A14	alternative_14	13.5	7.2	-22.6	96.1	2.0			
15	A15	alternative_15	-5.7	-118.3	-34.2	21.7	1.5			
16	A16	alternative_16	72.4	-185.9	-280.0	12.5	6.7			
17	A17	alternative_17	17.0	-34.6	-19.4	35.5	3.4			
18	A18	alternative_18	-31.2	-27.9	6.3	7.0	1.3			
19	A19	alternative_19	14.1	-48.2	6.8	16.6	1.6			
20	A20	alternative_20	60.6	-49.2	-17.2	7.2	0.3			
21	A21	alternative_21	26.2	-19.2	-36.7	90.4	0.8			
22	A22	alternative_22	7.0	-18.1	-6.5	16.5	0.9			
23	A23	alternative_23	-53.1	-98.0	-20.8	26.6	1.7			
24	A24	alternative_24	-17.2	-129.0	-14.2	267.9	1.3			
25	A25	alternative_25	32.7	-4.0	-15.8	177.4	2.1			
26	A26	alternative_26	26.7	-8.7	-36.3	32.5	2.8			
27	A27	alternative_27	-7.7	-59.2	-12.8	21.3	2.1			
28	A28	alternative_28	18.0	-13.1	-17.6	14.6	0.9			
29	A29	alternative_29	2.0	-38.0	1.6	7.7	1.2			
30	A30	alternative_30	-35.3	-57.9	0.7	13.7	0.5			
31	A31	alternative_31	5.1	-8.8	-9.1	100.9	0.9			
32	A32	alternative_32	0.0	-64.7	-4.0	0.7	0.1			
33	A33	alternative_33	25.2	-11.4	4.8	7.0	0.9			
34	A34	alternative_34	35.2	43.0	16.4	99.1	1.3			

The file has to be saved as a txt file, because Electre Tri only recognizes this kind of files to be loaded.

To import the data from a text file, click on the *"File>Import Alternatives..." button* and look for the file in your folder.

APPENDIX 3: PSEUDO-CODE IMPLEMENTATION OF THE ELECTRE TRI METHOD

- **<u>Credibility Calculation Method:</u> Returns credibilityAB vector and credibilityBA vector.**

```
Load(Alternative[] alternatives);
Load(Profile[] profiles);
Load(Criterion[] criteria);
Load(float[] weight);
Create(float[][] credibilityAB);
Create(float[][] credibilityBA);
Initialize(globalConcordanceAB=0);
Initialize(globalConcordanceBA=0);
AlternativesLoop(a from 0 to (numAlternatives-1)){
    ProfilesLoop(b from 0 to (numProfiles-1)){
        Initialize(concordanceABSum=0);
        Initialize(concordanceBASum=0);
        Initialize(weightSum=0);
        CriteriaLoop1(g from 0 to (numCriteria-1)){
            concordanceABSum = concordanceABSum + weight[g] *
            partialConcordance(alternatives[a],profiles[b],criteria[g]);
            concordanceBASum = concordanceBASum + weight[g] *
            partialConcordance(profiles[b], alternatives[a],criteria[g]);
            weightSum = weightSum + weight[g];
        }
        globalConcordanceAB = concordanceABSum/weightSum;
        globalConcordanceBA = concordanceBASum/weightSum;

        Initialize(credibilityABPlus=0);
```

```
            Initialize(credibilityBAPlus=0);


            CriteriaLoop2(g from 0 to (numCriteria-1)){
                If(partialDiscordance(alternatives[a],profiles[b],criteria[g])>
                globalConcordanceAB){
                    credibilityABPlus = credibilityABPlus * ( 1 -
                    partialDiscordance(alternatives[a],profiles[b],criteria[g])
                    / 1 – globalConcordanceAB);
                }
                If(partialDiscordance(profiles[b],alternatives[a],criteria[g])>
                globalConcordanceBA){
                    credibilityBAPlus = credibilityBAPlus * ( 1 -
                    partialDiscordance(profiles[b],alternatives[a],criteria[g])
                    / 1 – globalConcordanceBA);
                }
            }
            credibilityAB[alternative[a]][profiles[b]] = globalConcordanceAB *
            credibilityABPlus;
            credibilityBA[alternative[a]][profiles[b]] = globalConcordanceBA *
            credibilityBAPlus;
        }
}
```

Note: CriteriaLoop1 and CriteriaLoop2 calculate respectively globalconcordance and credibility to an alternative and profile.

- **<u>Relation Calculation Method:</u> Returns relation vector.**

```
Load(Alternative[] alternatives);
Load(Profile[] profiles);
Load(float[][] credibilityAB);
Load(float[][] credibilityBA);
Create(char[][] relation);
AlternativesLoop(a from 0 to (numAlternatives-1)){
    ProfilesLoop(b from 0 to (numProfiles-1)){
        if((credibilityAB[alternative[a]][profiles[b]] > λ )){ // aSb
            if((credibilityBA[profiles[b]][alternative[a]] > λ )){ // bSa
                relation[alternative[a]][profiles[b]] = 'I'
            }else{ //not bSA
                relation [profiles[b]] [alternative[a]] = '>'
            }
        }else{ // not aSb
            if((credibilityBA[profiles[b]][alternative[a]] > λ )){ // bSa
                relation[alternative[a]][profiles[b]] = '<'
            }else{ //not bSA
                relation [profiles[b]] [alternative[a]] = 'R'
            }
        }
    }
}
```

- **<u>Assignment Procedure Method:</u> Returns two vectors:**

  assignmentCategoriesPessimistic and assignmentCategoriesOptimistic.

  Load(Alternative[] alternatives);

  Load(Profile[] profiles);

  Load(float[][] credibilityAB);

  Load(char[][] relation);

  Create(char[][] assignmentCategoriesPessimistic);

  Create(char[][]assignmentCategoriesOptimistic);

  AlternativesLoop(a from 0 to (numAlternatives-1)){

  PessimisticLoop(b from (numProfiles-1) to 0){

  if((credibilityAB[alternative[a]][profiles[b]] > λ)){ // aSb

  assignmentCategoriesPessimistic[alternative[a]] = category[b]

  }

  Decrease(b = b -1);

  }

  OptimisticLoop(b=0; while relation[alternative[a]][profile[b]] != '>'){

  Increase(b = b +1);

  }

  assignmentCategoriesOptimistic[alternative[a]] = category[b]

<u>}</u>