

Domain Editor User Manual

W.Pasman, July 20, 2012

Introduction

The domain editor enables creation and modification of domain description for the pocket negotiator. In this document, "domain" is used as short for the domain description.

Installation

Make sure you have installed Java version 6. Higher versions are not compatible with the PocketNegotiator core. You can download the installer from <http://java.com/en/download/>. Some Java installers do not set up the system PATH variable properly to point to the java executable. Instructions to do this are found on <http://www.java.com/en/download/help/path.xml>.

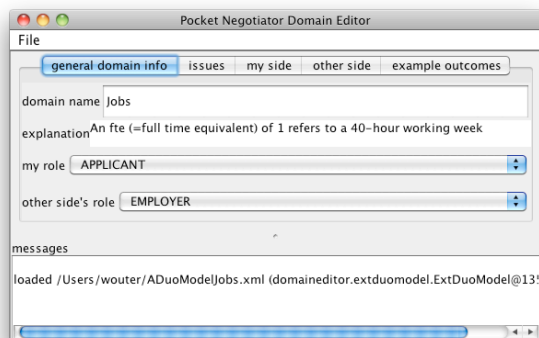


Figure 1. Domain Editor after loading the DUOModelJobs.xml file

General Workflow

Generally the work flow for editing profiles is like this

1. Start the editor by double clicking the **domaineditor.jar** file.
2. If you want to edit an existing domain, get it first on your local machine. using download.
3. Use the File/Open menu to open the domain that you want to edit. Figure 1 shows the screen after loading the DUOModelJobs.xml file. Or create a new one using File/New Domain.
4. Apply changes as discussed below.
5. Save the domain:

- a. using File/save which uses the **default filename format** which is "DUOModel<profile name>.xml" in your current working directory (usually the place where the domaineditor.jar file was runned).
- b. File/SaveAs allows you to choose your own file name. You may want to stick with the default format to ensure compatibility with the Pocket Negotiator.

6. Upload your saved domain (to your PNPrototype/resources directory)

Interest Profile structure

This section discusses the internal structure of a domain and interests.

It is assumed that there are two parties, or sides, involved in a negotiation: **my side** and **other side**. These are the two parties that are trying to reach an agreement. **my side** is the side that the user of the PocketNegotiator identifies with. **other side** is the party that the user is negotiating with.

The negotiation is about a number of **issues**: the concrete negotiable aspects such as price and quantity.

Each issue has different possible **issue values**.

Each side puts different **utility** into each issue value. A utility is a number between 0 and 1, and the higher the number, the more that side likes that particular issue value.

Each side has a number of **interests** or goals. The goals are different for each side, and therefore they put different **weights** and different onto the issues, and different importance on different **issue values**.

The interests, weights, issues and values make up a tree that is used to compute utilities of interests (see also **Figure 2**). These trees are called the **utility tree** of my side and other side.

Example

In this example we look in more detail to the job application domain.

In this domain, the **issues** (concrete negotiable aspects) are salary, fte (occupancy), work from home, lease car, permanent contract and career development opportunities. Career development opportunities has three concrete **issue values**: low, medium and high. Salary has five values: 2000, 2500, 3000, 3500 and 4000.

The **role** of my side here is that of applicant: the person using this profile is assumed to be soliciting to get a job (and not offering one). The other side's role is that of an employer offering the job.

In **my side's interests** we see the following (among others):

- wealth. This assumes that the user wants this job to achieve wealth. That reflects itself in a high **weight** of salary. And, a bigger salary is preferred over a small salary, so the issue value salary=4000 gets **utility**=1, and salary=2000 gets **utility**=0. Similarly, wealth reflects itself in a high **weight** for job occupancy: it is important to get a good amount of time on the job since $\text{income} = \text{occupancy} * \text{salary}$. Work from home on the other hand has a weight of zero for wealth.

- family life. This assumes that the user has a family and wants to spend time with that. This reflects itself in a high **weight** of work from home. And, more at home is better so gets high utility. Salary is not so important for family life, but occupancy is important here as well but for a very different reason here: part time jobs are better since that gives one more free time to spend with the family. Also family life is related to a permanent contract, since that gives him more stability for his family. Therefore, myside puts a higher utility on a *lower* occupancy.

For the other side we see the very different interests and utilities:

- team spirit. The employer is looking for a good and fast working team. Therefore he wants the applicant to be working with 100% occupancy, and to work at work instead of at home. The occupancy **weighs** more important to him than the work from home.

- low expense. The employer wants to do many things with his money so keeping salaries low is important. This reflects in preferring low salary and low occupancy. Economy is not going too well so he places more **weight** on the low expenses and less weight on team spirit.

Note that even within each side there are conflicts. For example the put high utility on high occupancy to get wealth, but also on the low occupancy to get family life.

All these factors are designed and given default values by the profile designer. The user of the pocket negotiator can only change part of the weights.

We can turn these profiles into a **utility tree** for each side, as in **Figure 2**.

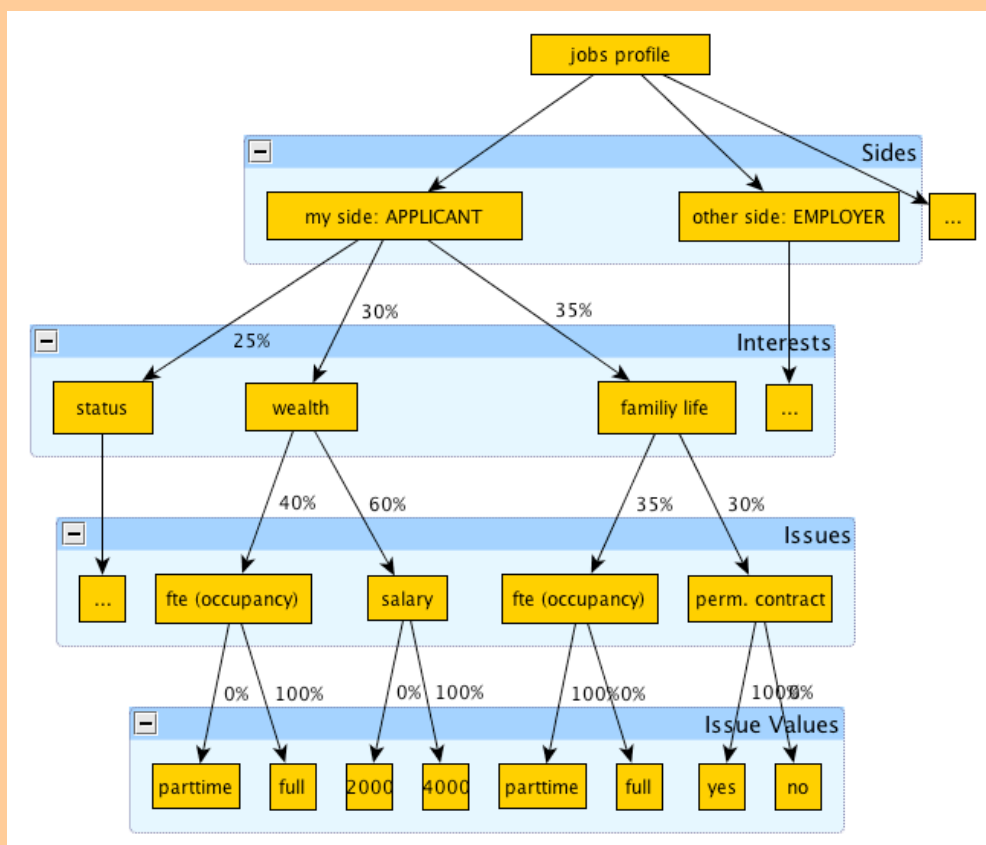


Figure 2. utility tree of my side, showing interests, issues, issue values, weights (top two rows of percentages) and utilities (bottom row of percentages).

We have two sorts of interests for each side: the 'normal' and the 'real' interest in the domain. The normal interest is the estimated interest that is used as a starting point. The real interest

is a theoretic 'optimal' value and is used only for comparison afterwards in training and for steering of an opponent bot in the pocket negotiator.

Upload and Download

For upload and download, you need a connection to the internet.

Download

Use the File/Download menu. Select a domain using the combo box and click OK. The file will be saved in the same directory where your domaineditor.jar file is.

Upload

After you have saved an edited profile, you have to upload it to the server so that you can use it for an actual negotiation. For upload, your edited domain file must have a name that is not yet in use. You can check if a name is not in use by using the Download option and checking that your name is not in the list. Domain names can contain only letters (upper or lower case) and numbers. Names are case sensitive.

To upload, use the File/Upload menu. Select the fixed domain file and click "Open".

Editing general domain info

To edit the general domain info, select the general domain info tab (Figure 3). Here you can change the following

- domain name: type a short name for the domain. This text will be used in sentences like "You selected negotiating a NAME contract" where NAME is the short name that you give here.
- description. This text will be shown on the preparation page of the PocketNegotiator to help the user with domain-specific explanations.
- my role. A combo box allows you to select the proper role for my side
- other side's role. This combo box allows you to select the proper role for the other side.

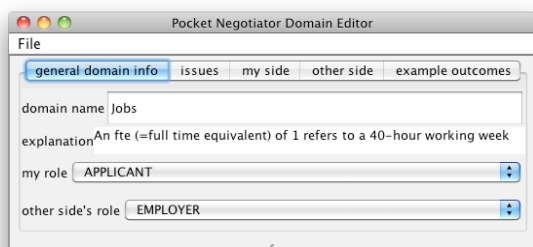


Figure 3. The general domain info tab.

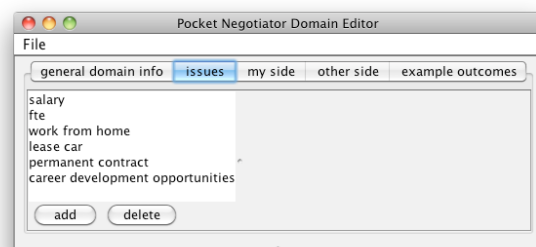


Figure 4. The issues tab.

Editing the issues

To edit the issues, select the issues tab (Figure 4). From this point you can add or remove issues, or change issue values.

Add an issue

Add an issue by pressing the Add button. Select the type of the issue from the appearing dialog combo box. There are three types available: numeric, yesno and nominal. numeric type is used to represent numeric issue values. The nominal type is used to represent a set of textual alternatives, such as 'red', 'green' and 'blue' for the issue preferred color. yesno type is used for nominal types with a 'yes' and 'no' option.

Delete an issue

To delete an issue, select the issue in the list of issues and press the delete button. **Warning:** deleting an issue will also delete all occurrences of the issue from the utility trees.

Edit an issue

To edit an issue, select the issue in the list of issues. An issue-value editor matching the type of the issue appears in the right side.

- numeric issue (Figure 5). The numeric issue name, minimum, maximum and step size can be edited. To change the name, click in the name and type.
- nominal issue (Figure 6). The issue name and the issue values can be changed. To change the name, click in the name and type. To remove a value, select the to be removed value and select the delete button **under the editor**. Note that there is another delete button under the issues, but that one will delete the issue altogether. To add a new value, press the add button. To change the name of value, double click on the value.
- Yesno issue (Figure 7). This issue always has only the values 'yes' and 'no' and you can only change the name of the issue.

In all issue value editors, changes to an existing issue value or to the name will be propagated through utility tree. But changing the available values does not change the utility trees. This is because there is no clear default way to deal with insertion or removal of values. Therefore, you may have to edit the utility trees when you change the set of issue values.

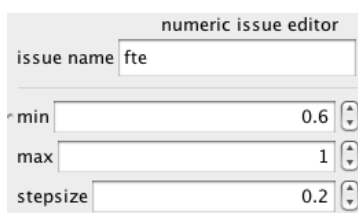


Figure 5. Numeric issue editor when the fte issue is selected.

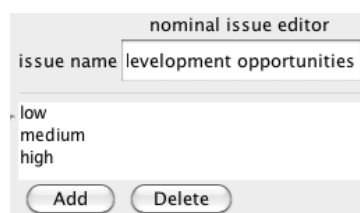


Figure 6. Nominal issue, when the carreer development issue is selected.

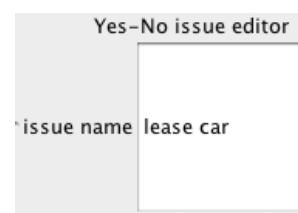


Figure 7. Yes-no issue, when the lease car issue is selected.

Editing a Side

To edit the main properties, select the 'my side' or 'other side' tab (Figure 8). This section discusses the 'my side' case only, as the 'other side' works identically.

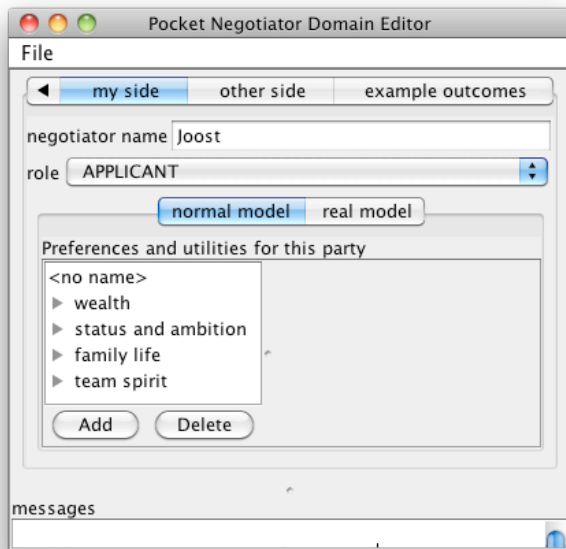


Figure 8. Main area of the my side. At the bottom left is the actual (folded) utility tree for this party.

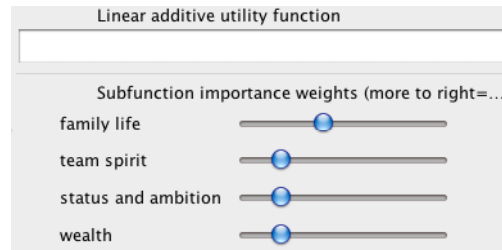


Figure 9. sliders to set the interest weights after selecting the root node.

You can change the negotiator name by clicking on the name and typing.

You can select the role by picking the required role from the combo box.

The part below "Preferences and utilities for this party" are for editing the actual utility tree and is discussed in the next subsections.

Editing interest weights

To change the interest weights, select the root (<no name> in Figure 8). The linear additive utility function editor appears for the root node (Figure 9). Use the sliders to set the weight of each of the interest nodes under the root node between 0 (left) and 1 (right).

Add or remove an interest

To add an interest to the utility tree, select the root (<no name> in Figure 8). Then press "Add". A new, empty interest is inserted and you need to add issues to it as explained below.

To remove an interest, select the interest and press the delete button.

Add or remove an issue

To add an issue to an interest, select the interest and press "Add". A dialog pops up (Figure 10). Select the issue to be added from the combo box and press OK.

To remove an issue, select the issue and press delete.

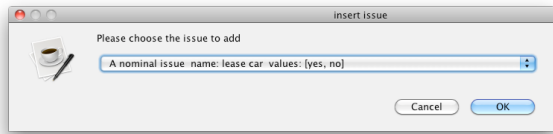


Figure 10. Dialog to request the new issue.

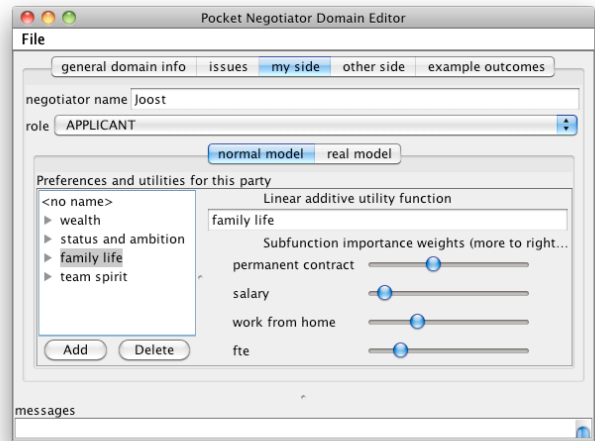


Figure 11. After an interest (here, 'family life') is selected, a linear additive utility function editor appears, with sliders for each of the issues for this interest.

Change issue weights

To change the weights of issues inside an interest, select the interest. A linear additive utility function editor appears with a slider for each of the issues contained in this interest. Use the sliders to set the issue weight between 0 (left) and 1 (right).

Changing issue utilities

To change the utilities associated with possible values of an issue, select the issue. You may have to unfold the utility tree a bit to select the issue, as they are inside the interests. Do this by clicking on the triangle left of the interests.

When the issue is selected, the issue utility function editor suited for the issue will appear (Figure 12).

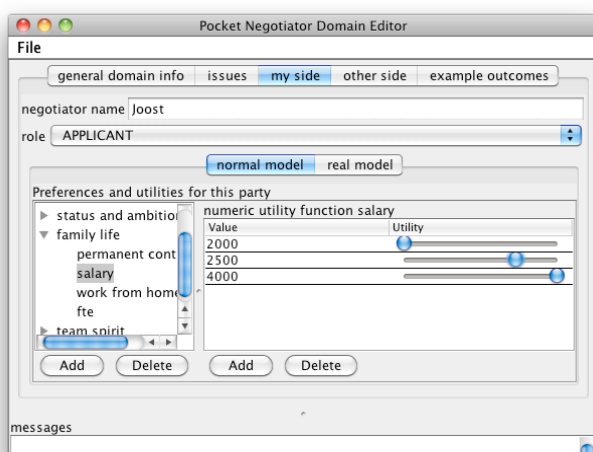


Figure 12. An issue-value utility editor appears when an issue is selected in the utility tree. Here, a numeric issue was selected.

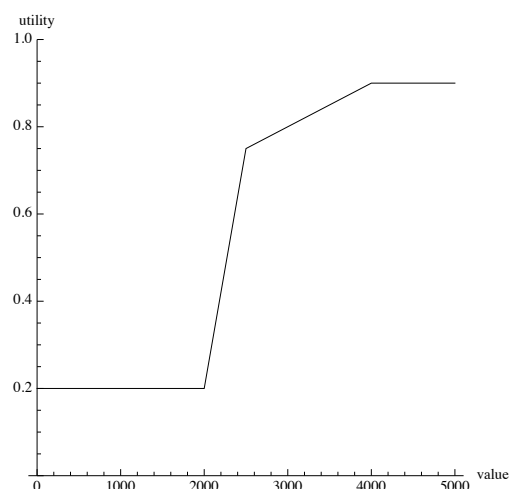


Figure 13. Value-utility graph given the points for utility of salary.

Numeric issue-value editor

When a numeric issue is selected, the numeric issue-value editor appears (Figure 12). You drag the sliders to the right to increase the utility of a value for that utility, and to the left to decrease the utility.

To explain the meaning of the numeric value-utility pairs, we have to explain how these are used by the pocket negotiator. When the pocket negotiator needs to evaluate the utility of a numeric value for this utility, it does a linear inter/extrapolation of the given points. To illustrate this, consider the three points as in Figure 12: $\langle 2000, 0.2 \rangle$, $\langle 2500, 0.75 \rangle$ and $\langle 4000, 0.9 \rangle$. This means that for example a salary of 2500 would have a utility for Joost of 0.75. To compute the utility of a salary of points not in the list, for example 3000, linear interpolation/extrapolation is done as in Figure 13: between known points, the values are interpolated, and outside the range the utility is clamped to the nearest known point.

To add new points, press the Add button at the bottom of the numeric issue-value editor. A popup appears asking for the new value to be added. Note that you can only pick the values defined by the minimum, maximum and step size as set for the issue. The default values for a numeric issue are utility=0 at the minimum value and utility=1 at the maximum value.

To remove a value, select the value and press the Delete button.

Use the sliders to set the utility of each of a value between 0 (left) and 1 (right).

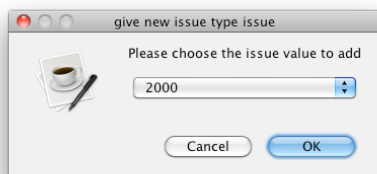


Figure 14. Popup asking for a new issue value.

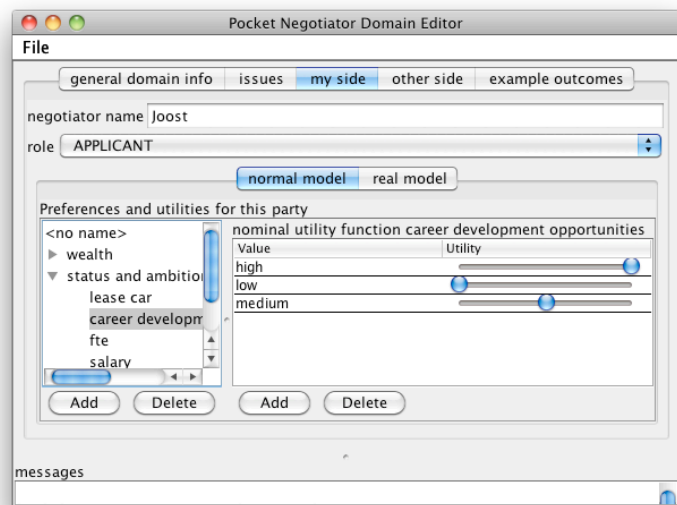


Figure 15. Nominal issue-value editor appears when a nominal issue (or yes-no issue) is selected.

Nominal issue-value editor

If a nominal issue or yes-no issue is selected, the nominal issue-value editor appears (Figure 15). In the nominal issue-value editor no interpolation is used. Instead, values that are not in the list are assumed to have utility=0.

To add a value-utility pair, press the Add button. A popup similar to Figure 14 appears, allowing you to select the issue value to be added.

To remove a value-utility pair, select the value and press Delete.

Use the sliders to set the utility of each of a value between 0 (left) and 1 (right).

Editing example outcomes

To view or edit the example outcomes, select the example outcomes tab. To see the details of an outcome, select the outcome from the example outcomes list (Figure 16).

To add a new outcome, press the Add button below the example outcomes list. To remove an outcome, select the outcome in the example outcomes list and press Delete.

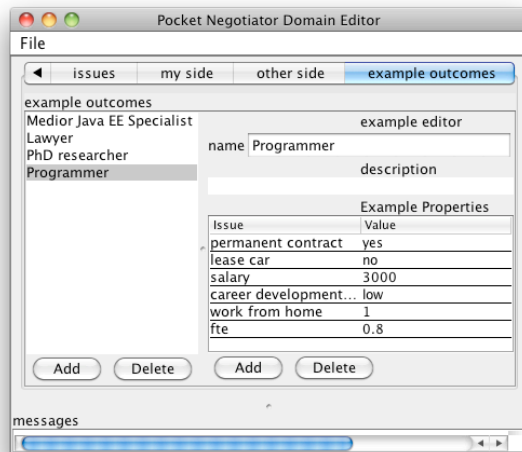


Figure 16. The example outcome panel. The Programmer example outcome has been selected and its details are shown in the right part of the panel.

Change the outcome

To change the name of the example outcome, click in the name field and type the required changes.

To change the description of the example outcome, click in the description field and type the required changes. Note that the words may brake at weird places. The pocket negotiator uses a different algorithm for word breaking, so it should look all fine once viewed in the pocket negotiator.

To change the issue-value properties of the outcome, click on the value that you want to change. A combo box appears that allows you to select the required value. Note that for numeric issues, the available values are not continuous but determined by the minimum, maximum and step size of the selected issue.

To add a new issue-value, press the Add button under the example properties. A dialog as in Figure 10 appears to request the new issue to add.

To remove an issue-value pair, select the issue and press Delete.