

# OWLS-XPlan

OWL-S Semantic Web Service Composition Planner

Version 1.0



## User Manual

February 20, 2006

Andreas Gerber, Suleman Butt, Matthias Klusch  
(c) DFKI Saarbrücken, Germany

## Table of content

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>INTRODUCTION.....</b>                          | <b>3</b>  |
| <b>2</b> | <b>INSTALLATION.....</b>                          | <b>4</b>  |
| <b>3</b> | <b>STARTING OWLS-XPLAN.....</b>                   | <b>5</b>  |
| <b>4</b> | <b>HOW TO USE OWLS-XPLAN?.....</b>                | <b>6</b>  |
| 4.1      | Preparation.....                                  | 7         |
| 4.2      | Planning.....                                     | 9         |
| 4.3      | The PDDXML Editor.....                            | 11        |
| 4.4      | QoS Driven Optimization of Composition Plans..... | 19        |
| <b>5</b> | <b>DISCLAIMER.....</b>                            | <b>30</b> |
| <b>6</b> | <b>SUPPORT AND CONTACT.....</b>                   | <b>30</b> |

# 1 Introduction

This document provides information on how to install and use the OWL-S service composition planner OWLS-XPlan version 1.0. In particular, it explains the various functional cards or tabs of its graphical user interface, each of which related to the preparation and execution of the planning process. Using these cards to proceed through the GUI from left to right, directly corresponds to the usual sequence of the service composition planning process.

We acknowledge that this initial version is for expert users only. In particular, it is assumed that the user is sufficiently familiar with using OWL, OWL-S, and PDDL.

Upcoming releases will provide the novice user with an extended GUI extended by more convenient facilities for (semi-)automatically generating the initial state and goal state ontologies that are required as an input to the planning process, and querying OWLS-XPlan.

OWLS-XPlan allows for fast and flexible composition of OWL-S services in the semantic Web. OWLS-XPlan converts OWL-S 1.1 services to equivalent problem and domain descriptions that are specified in the planning domain description language PDDL 2.1, and invokes an efficient AI planner XPlan to generate a service composition plan sequence that satisfies a given goal. XPlan extends an action based FastForward-planner with a HTN planning and re-planning component.

For more detailed information on OWLS-XPlan, we refer to the Readme.txt and relevant publication:

Klusch, M.; Gerber, A.; Schmidt, M. (2005). Semantic Web Service Composition Planning with OWLS-XPlan. Proceedings of the 1<sup>st</sup> Intl. AAAI Fall Symposium on Agents and the Semantic Web, Arlington VA, USA, AAAI Press, 2005

The **OWLS-XPlan project web page** is <http://www.dfki.de/~klusch/owl-s-xplan>

OWLS-XPlan project @semwebcentral.org:  
<http://projects.semwebcentral.org/projects/owl-s-xplan/>

## 2 Installation

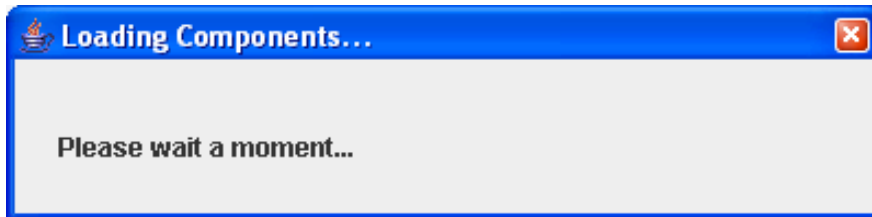
Installing the OWLS-XPlan software is as easy as follows:

1. Please download the distributed **owlsxplan.zip** file from the project homepage
2. **Unzip this file into a local directory** <owlsxplan-dir>. We assume the name of this directory to be c:\OWLS-XPlan\_1.0.  
If you copy it to a differently named directory you then have to edit the java paths manually. The path descriptions used by the software are stored in the batch file **XPLAN.bat** that is stored in the zipped directory OWLS-XPlan\_1.0\src\  
Replace all paths in *XPLAN.bat* beginning with C:\OWLS-XPlan\_1.0 by the correct path on your local machine to the directory <owlsxplan-dir>.

Please check and, if necessary, correct the path to the JAVA directory on your local machine, such as C:\Programme\Java\jdk1.5.0\_01\bin, in the batch file **XPLAN.bat**

### 3 Starting OWLS-XPlan

Start the batch file **OWLS-XPlan\_1.0\src\XPLAN.bat** by means of a double click on its icon, which results in loading of the components of the OWLS-XPlan package



... followed by the initial welcome screen

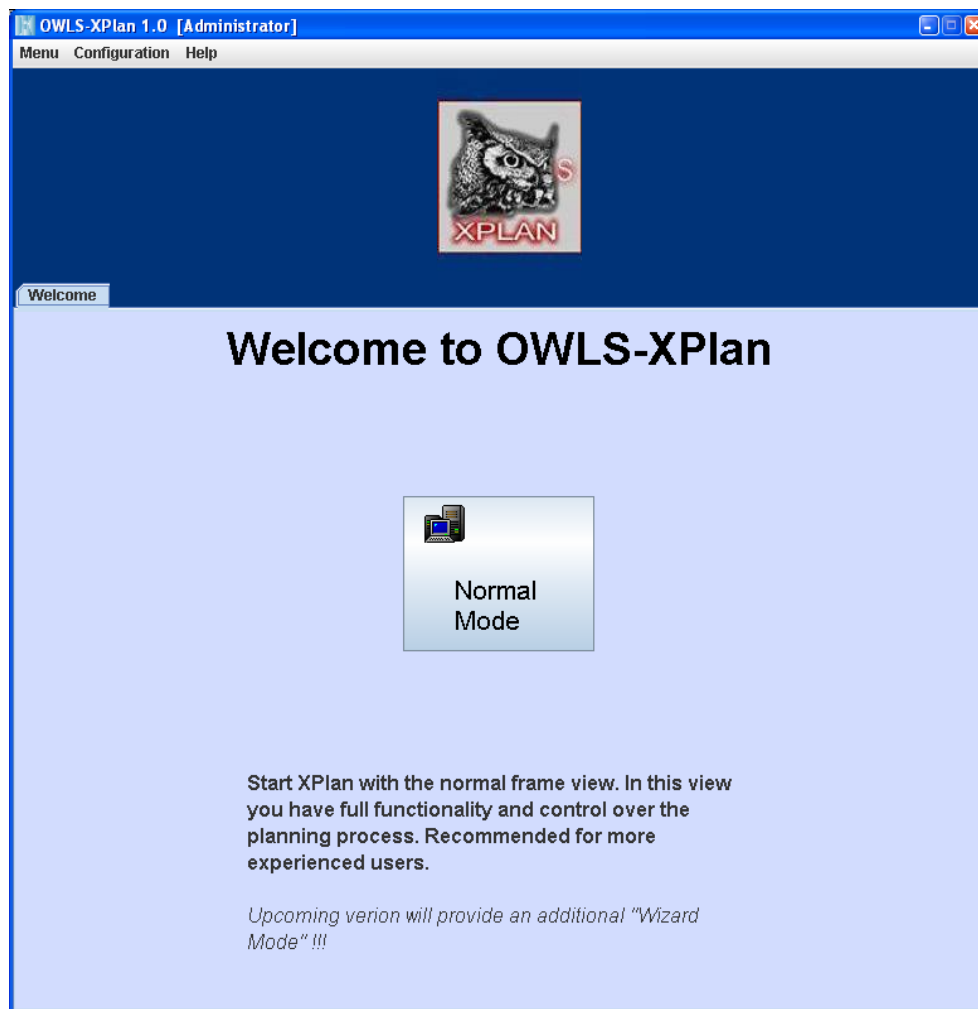


Fig.1: Welcome Screen

This initial release 1.0 of OWLS-XPlan does not provide the planned wizard mode. As a consequence, please click on the button **normal mode** to use the system.

## 4 How to Use OWLS-XPlan?

After starting OWLS-XPlan in the normal mode (expert level), you may choose between activities related to the “Preparation” of the planning process, and/or the “Planning” process itself: (Fig. 2)

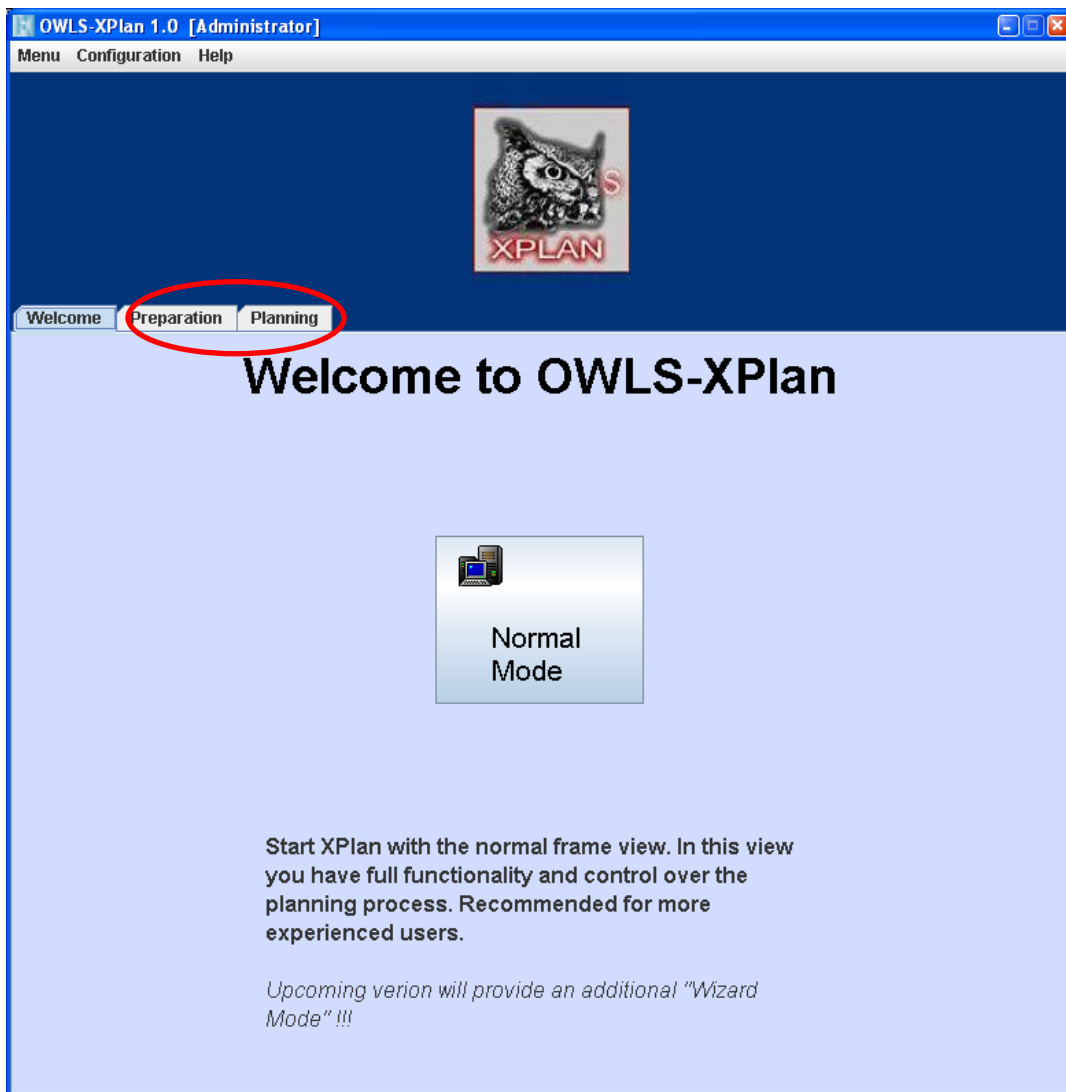


Fig. 2: Welcome Screen with all tabs related to the planning process

We will briefly explain the use of both respective tabs in subsequent sections.

## 4.1 Preparation

Using OWLS-XPlan in the normal mode for expert users requires both an initial state ontology and a goal state ontology as input. Please specify the location of the respective files by clicking on the *preparation* tab (Fig. 3).

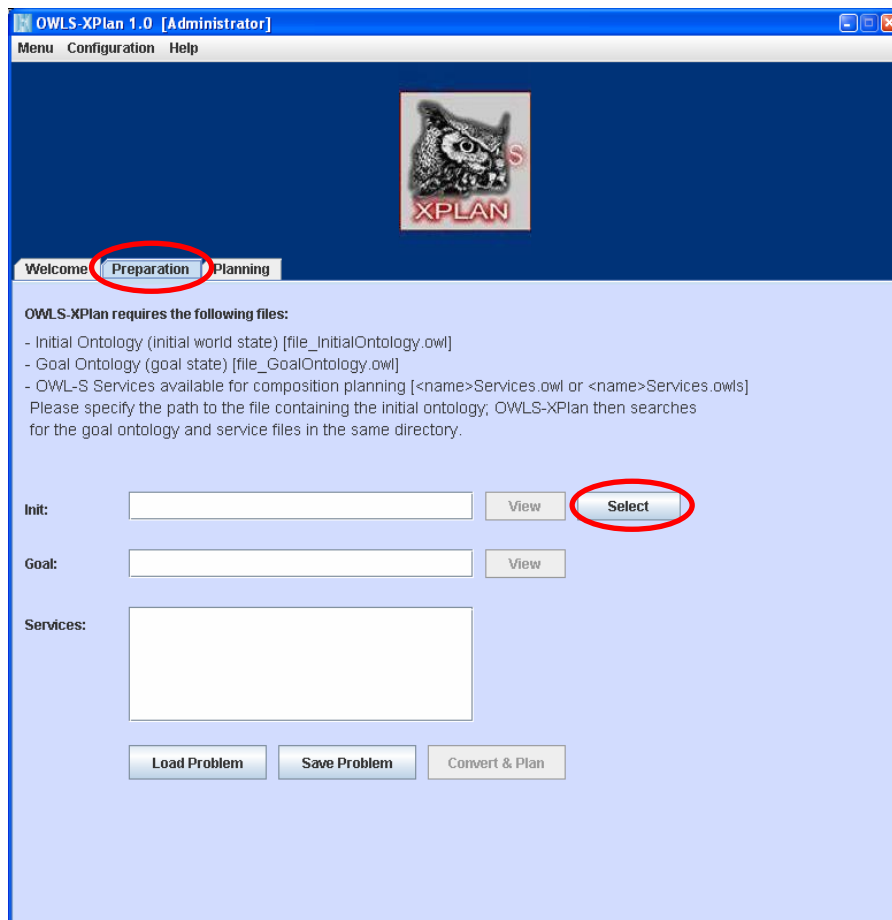


Fig. 3: Preparation: Select initial state and goal ontologies

### **EXAMPLE:**

You may use the initial state and goal ontologies provided in the distribution package as an example for using OWLS-XPlan. For this purpose, please specify the directory in which you have stored the OWL-S descriptions of the semantic web services, and then click on the select button which causes a file selector window to show up. Use it to select an initial ontology - our example ontology file in the health care domain is

**C:\OWLS-XPlan\_1.0\OWLS-Dateien\Health-Scallops\_InitialOntology.owl**

After this file has been selected the remaining goal state ontology file in OWL and the OWL-S service files are automatically searched for in the same directory, scanned, and attached (Fig. 4).

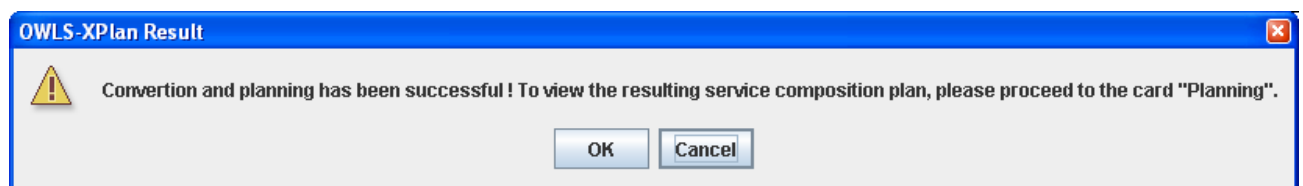


Fig.4: Preparation: Manual selection of initial state ontology file, automatic selection of corresponding goal state ontology and services in the same directory

- **load problem** : Loading of a planning problem in terms of the paths to the corresponding ontology files of your choice.
- **save problem** : Store the paths to the ontology and service files as provided in a separate file; this is to avoid having to repeatedly enter the paths.

### Conversion to PDDXML

To proceed to the conversion and planning process, please click on the **convert & plan** button. As a result, first, the OWL-S services are converted into PDDXML, and the planner is automatically invoked. If a plan could be successfully generated a status window appears:





If you are pressing on the **OK** button a new tab *optimization* will appear, and the screen changes to the *planning* tab.

## 4.2 Planning

The right side of the panel of the planning tab displays the visualization of the generated plan, while the left side of the panel shows the paths of both created PDDXML files (Fig. 5):

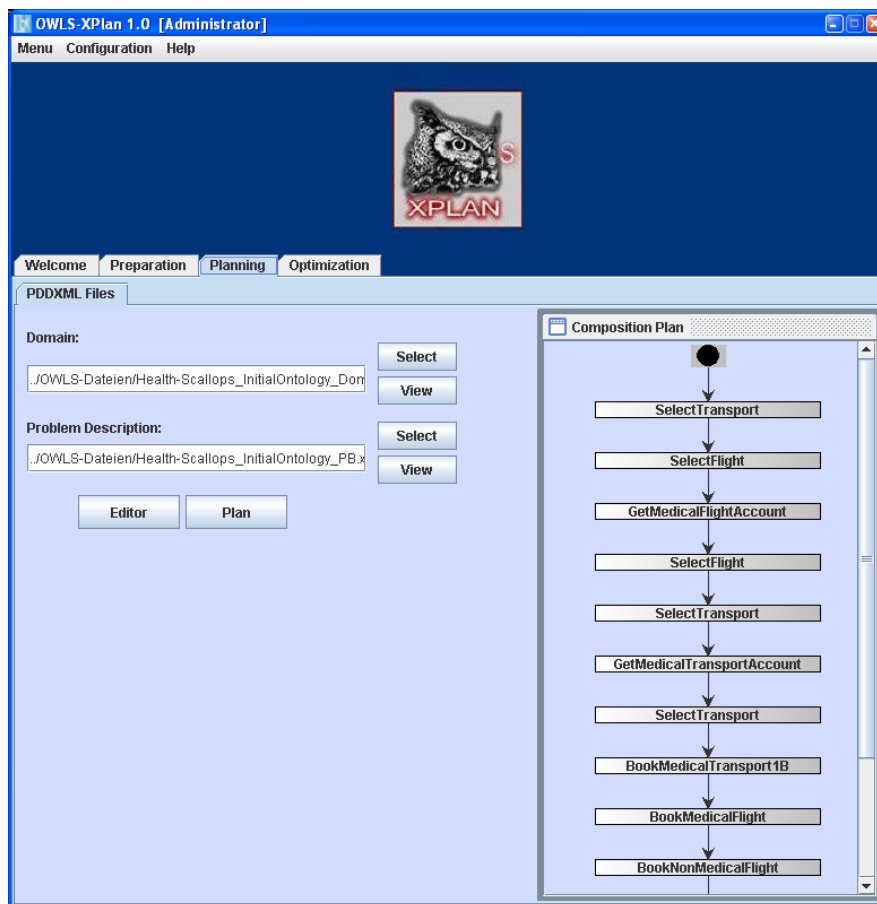


Fig.5: Planning: PDDXML file operations and visualization of the generated plan

- **Select** : Select a PDDXML file that contains the domain or planning problem definition  
(e.g. C:\OWLS-XPlan\_1.0\OWLS-DateienHealth-Scallops\_InitialOntology\_Domain.xml)
- **View** displays the structure of the content of the selected PDDXML file in a special browser (Fig.6). Changing of the content is not possible in this browser.

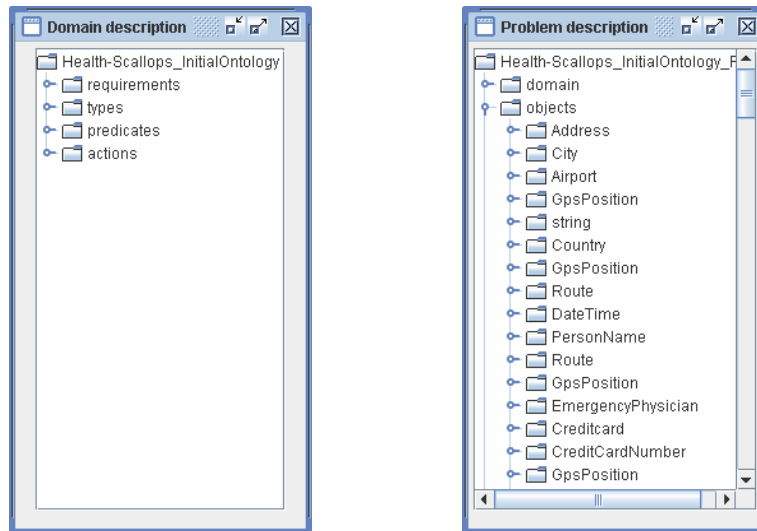


Fig.6: Browsing PDDL file content

- **Editor** invokes an integrated PDDL editor (Fig. 7) that allows you to edit the planning domain and problem description files

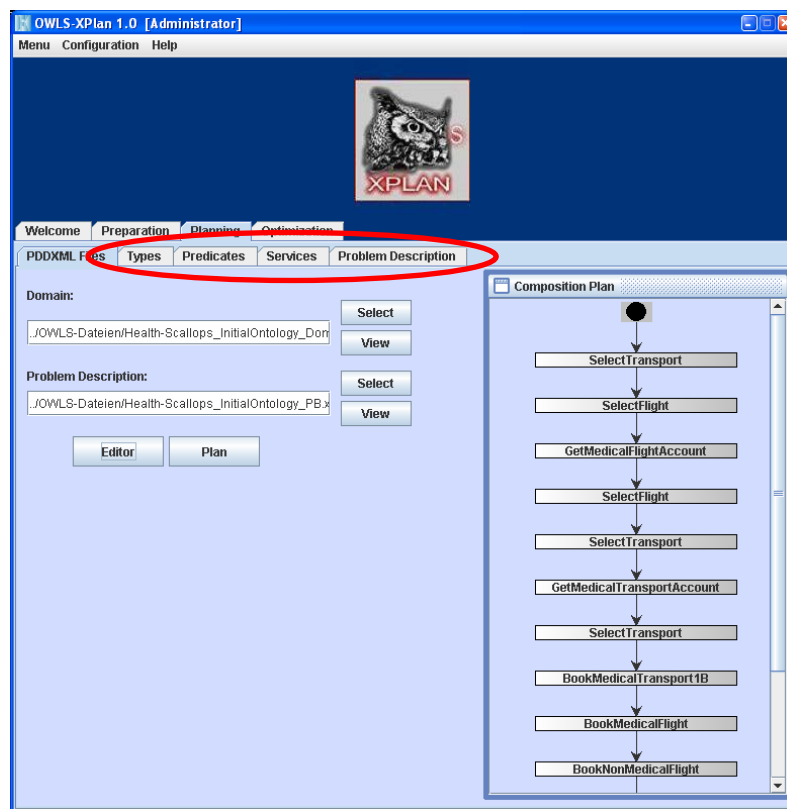


Fig.7: PDDL Editor

### 4.3 The PDDXML Editor

The tabs of the PDDXML editor correspond to all options of describing a planning domain and problem in PDDL. You can use the editor, for example, to change the content of the converted services and ontology files in PDDXML, or to create such files from scratch (though this is an admittedly tedious task to do for the novice user).

#### Types

The AI planner XPlan of the composition planner OWLS-XPlan uses a typed domain described by a hierarchy of (super-) types or concepts.

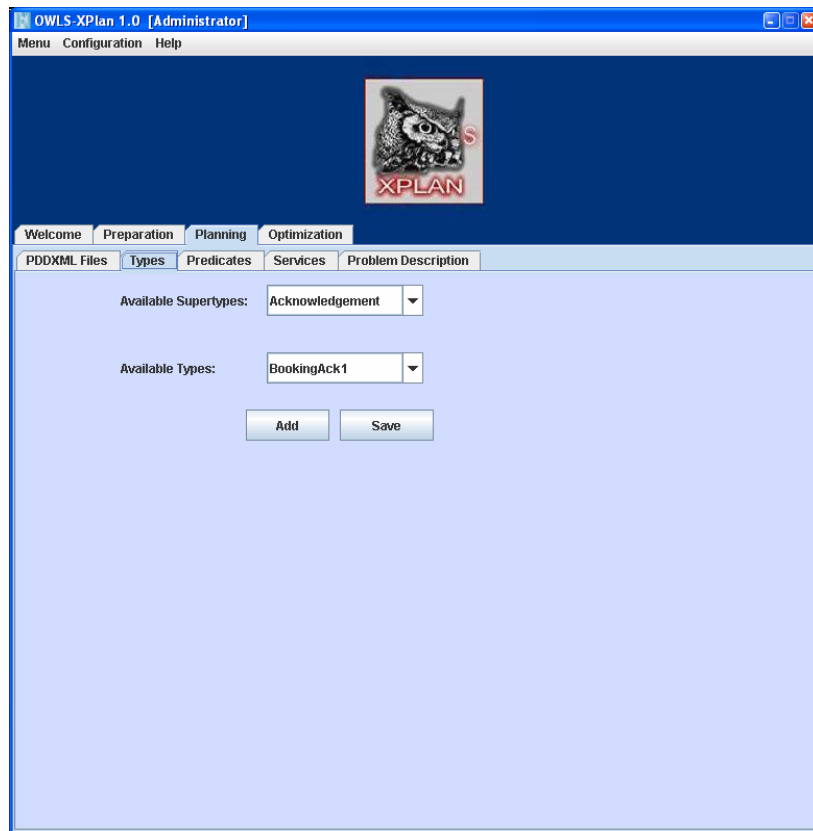
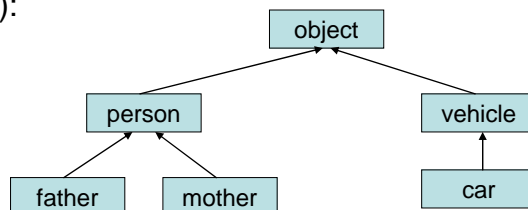


Fig.8: PDDXML Editor: Types

You may generate a complex type hierarchy such as the following (object is the root node of the hierarchy):



- **Available Super types** : If a super-type is selected in the combo-box, you can view all of its subtypes in the combo box **Available types**.

- **Add** : Create a new type (Fig.9) by selecting the type of which the newly created type is subtype of. Enter the (unique) name of this new type into the text field "Type Name". With **Add Type** the new type is included into the type hierarchy.

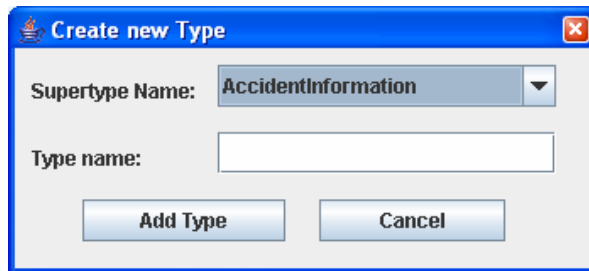


Fig.9: PDDXML Editor: Types - Add a new type to your type hierarch

Please save all the changes you did in the type hierarchy by pressing the **Save** button such that the planner can take them into account at the next time of its invocation when only reading the PDDML files as input.

### **Predicates**

If you want to change some predicate definition in the planning domain, you may use the *predicates* tab (Fig. 10)

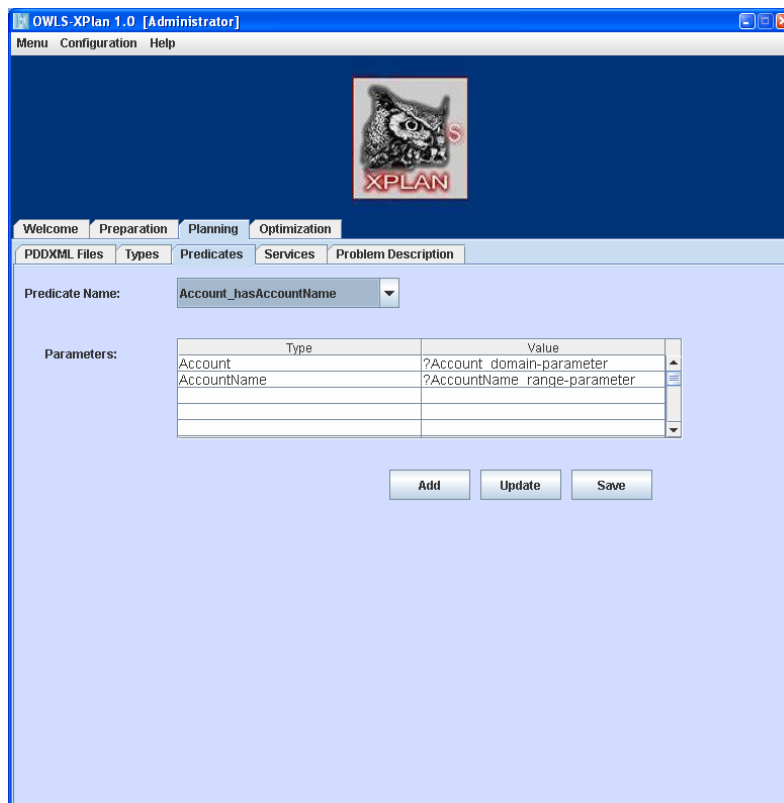


Fig. 10: PDDXML Editor - Predicates

In the combo box *predicate name*, you can either define a new, or select a predicate of the actually considered planning domain description. In the *parameters* box below, the parameters of this predicate are displayed (Fig. 11).

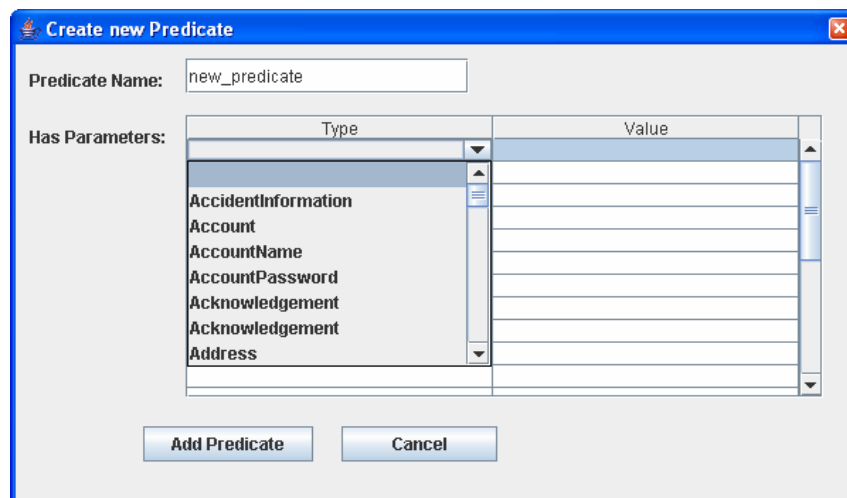


Fig. 11: PDDXML Editor: Predicates - Create a new / Modify existing predicate

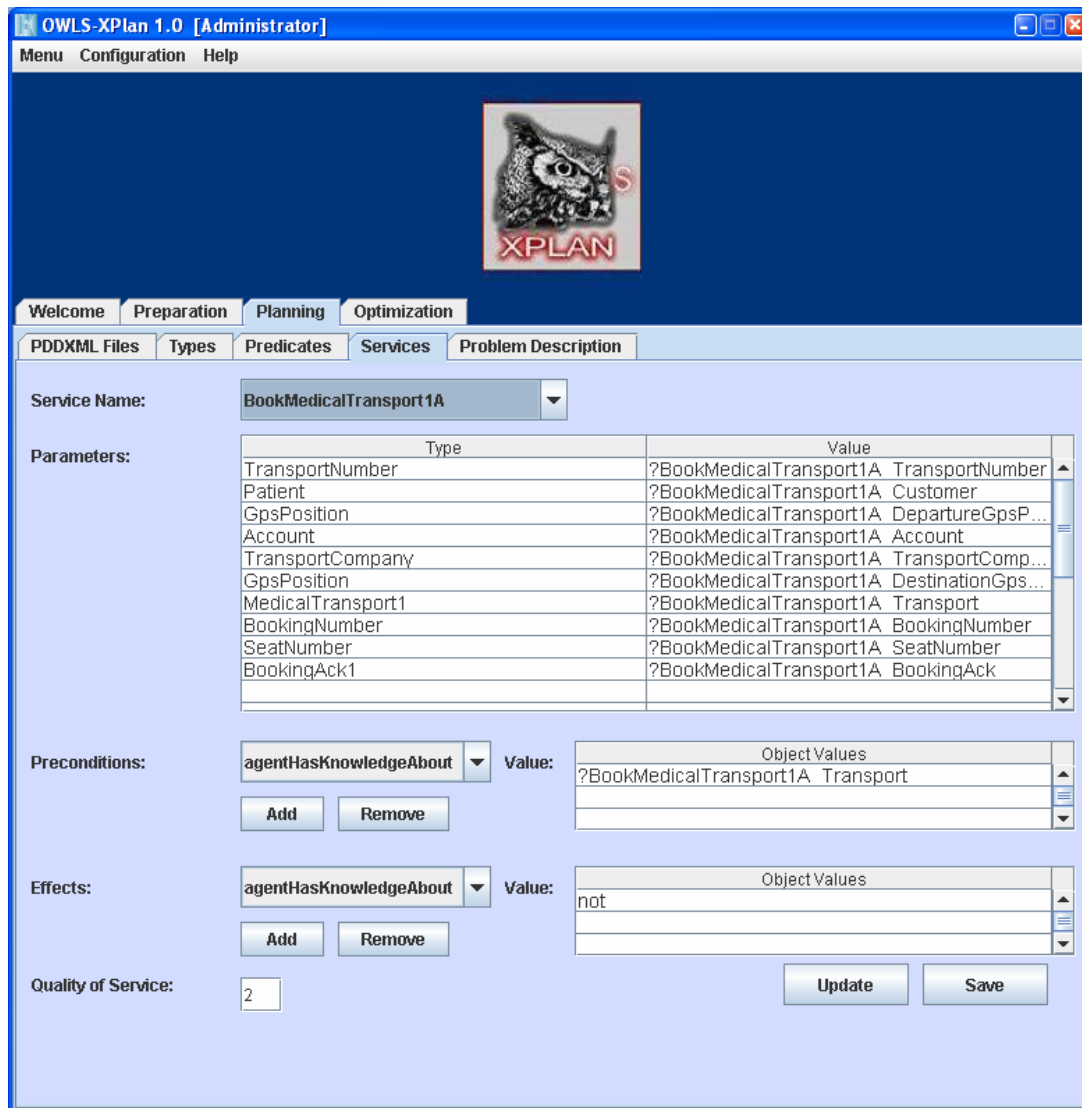
You can create or modify the definition of a selected predicate by clicking on the type field.

- **Type:** A combo box shows all available types of the parameters of the selected predicate. Each parameter name can be changed by a double click on the field: You may enter any string that is unique for the definition of the selected predicate.

**Save & Update:** After your work on the predicates is done, please click on both the *Save* and the *Update* button (Fig. 10) to save the changes in the respective PDDXML files, and propagate them to the planner, otherwise the changes will be discarded once you are selecting another predicate.

## Services

You can change the definition of any of the OWL-S services that are associated with the actual planning domain and problem by clicking on the *services* tab (Fig. 12).



The screenshot shows the OWLS-XPlan 1.0 Administrator interface. The window title is "OWLS-XPlan 1.0 [Administrator]". The menu bar includes "Menu", "Configuration", and "Help". The main area has a dark blue background with an owl logo and the text "XPLAN". Below the logo are tabs for "Welcome", "Preparation", "Planning", and "Optimization". Under "Planning", there are sub-tabs for "PDDXML Files", "Types", "Predicates", "Services", and "Problem Description". The "Services" tab is active, showing the configuration for a service named "BookMedicalTransport1A".

**Service Name:** BookMedicalTransport1A

**Parameters:**

| Type              | Value                                     |
|-------------------|---|
| TransportNumber   | ?BookMedicalTransport1A TransportNumber   |
| Patient           | ?BookMedicalTransport1A Customer          |
| GpsPosition       | ?BookMedicalTransport1A DepartureGpsP...  |
| Account           | ?BookMedicalTransport1A Account           |
| TransportCompany  | ?BookMedicalTransport1A TransportComp...  |
| GpsPosition       | ?BookMedicalTransport1A DestinationGps... |
| MedicalTransport1 | ?BookMedicalTransport1A Transport         |
| BookingNumber     | ?BookMedicalTransport1A BookingNumber     |
| SeatNumber        | ?BookMedicalTransport1A SeatNumber        |
| BookingAck1       | ?BookMedicalTransport1A BookingAck        |

**Preconditions:** agentHasKnowledgeAbout Value: Object Values  
?BookMedicalTransport1A Transport

**Effects:** agentHasKnowledgeAbout Value: Object Values  
not

**Quality of Service:** 2

Buttons: Add, Remove, Update, Save

Fig.12: PDDXML Editor - Services

To change the *parameter* list, please click in the "Type" field and select a new type from the appearing combo box. The *value* can be changed by a double click into the text field. Analogous to the changes of predicates, you can modify the list of *preconditions* and *effects*.

Again, please *update* and *save* your changes as described above.

## Problem Description

The second major component of a planning problem is, of course the problem description itself. You can modify your loaded problem description by clicking on the *problem description* tab.

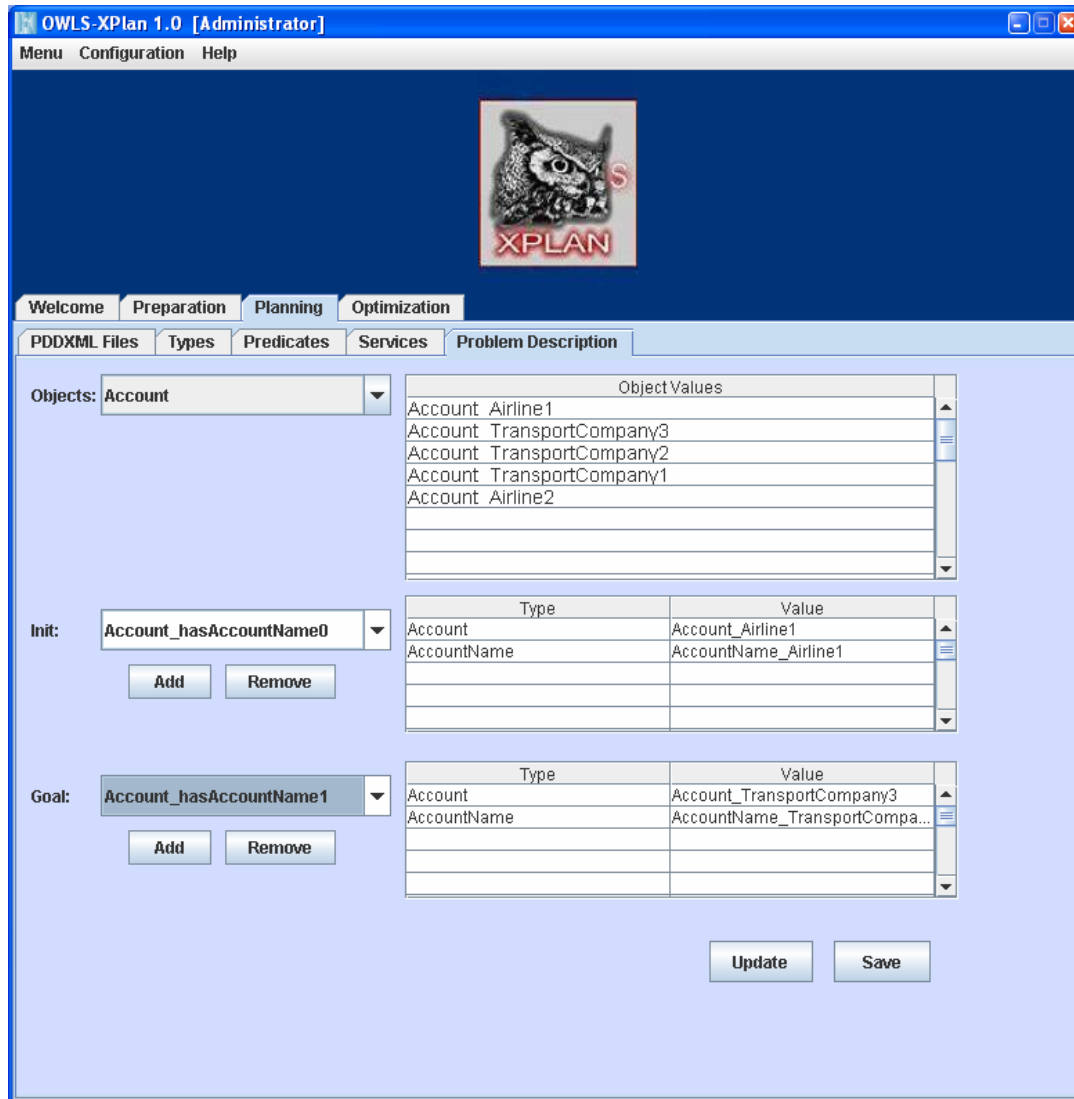


Fig.13: PDDXML Editor: Problem Description

- **Objects** : The *objects* combo box allows you to select a type of which all objects are displayed on the right side. By double clicking in the text field “Object Values” you can modify their name. The names have to be unique.
- **Init & Goal** : Analogous to the *predicate* tab, you can change the description of the *init* and *goal* state, by selecting, adding and removing predicates. You can only change the value of displayed parameters of the selected predicate. A combo box with all relevant objects of the selected type is displayed.

Please update and save your changes as described previously.

### Example in the Health-Care Domain:

Fig 14 shows the original composition plan generated by the OWLS-XPlan (based upon the existing Problem Description – C:\OWLS-XPlan\_1.0\OWLS-DateienHealth-Scallops\_InitialOntology\_PB.xml) is visualized under the *PDDXML Files tab*.

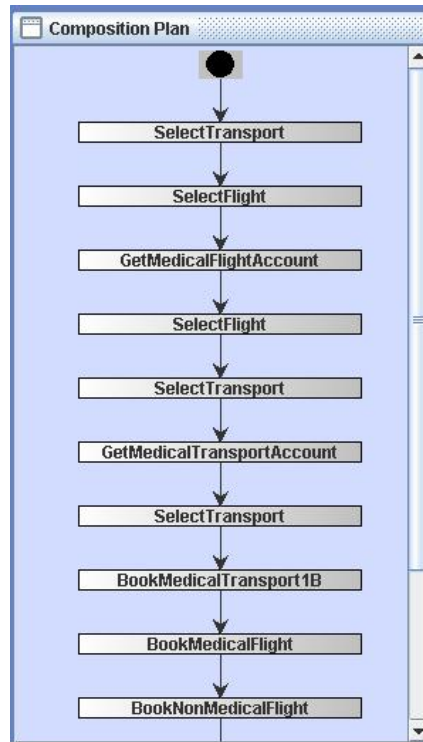


Fig.14: Example Service Composition Plan

Because using an AI-planner the user is able to make some changes in the PDDXML problem description. For example the user wants a different location the patient has to be brought. This change can be done under the *Problem Description tab*. In the original composition plan, the position in the goal state description of a person (e.g. *Patient\_Mikka*) is at some hospital (*Position\_Hospital2*) and should be changed to (*Position\_AirportFaro*), see below in (Fig. 15).



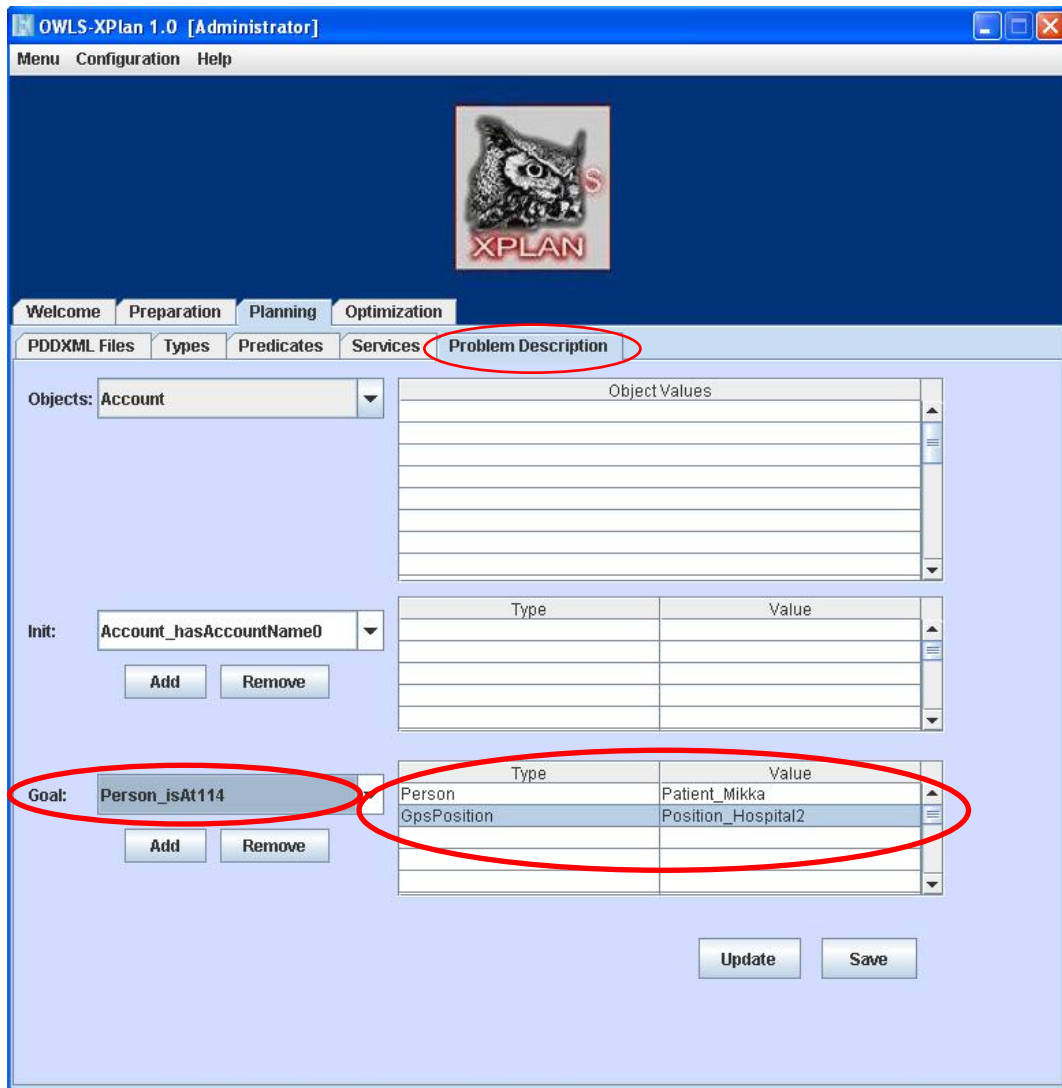


Fig.15: Problem Description tab

To change the destination of Mikka in the goal state, please first select the relevant predicat, called *Person\_isAt114*. The value for the destination position (*GpsPosition*) for Person (*Patient\_Mikka*) has then to be changed from *Position\_Hospital2* to e.g. *Position\_AirportFaro*.

To do so, please select the appropriate Goal state attribute (in this case, *Person\_isAt114*) from the combo box appearing on the left. Once the attribute is selected, the corresponding type/value pair(s) will appear in the Goal state table, on the right.

See figure 16. In order to change the *GpsPosition* value, click on the *Position\_Hospital2* value field. A combo box appears. In our example at the fourth position you will find *Position\_AirportFaro*. Select this value and press the *Save* button on the lower right corner. Now the global positioning value *Position\_Hospital2* for the person *Patient\_Mikka* has been changed to *Position\_AirportFaro* under the Goal state attribute *Person\_isAt114*. See below in (Fig. 16).

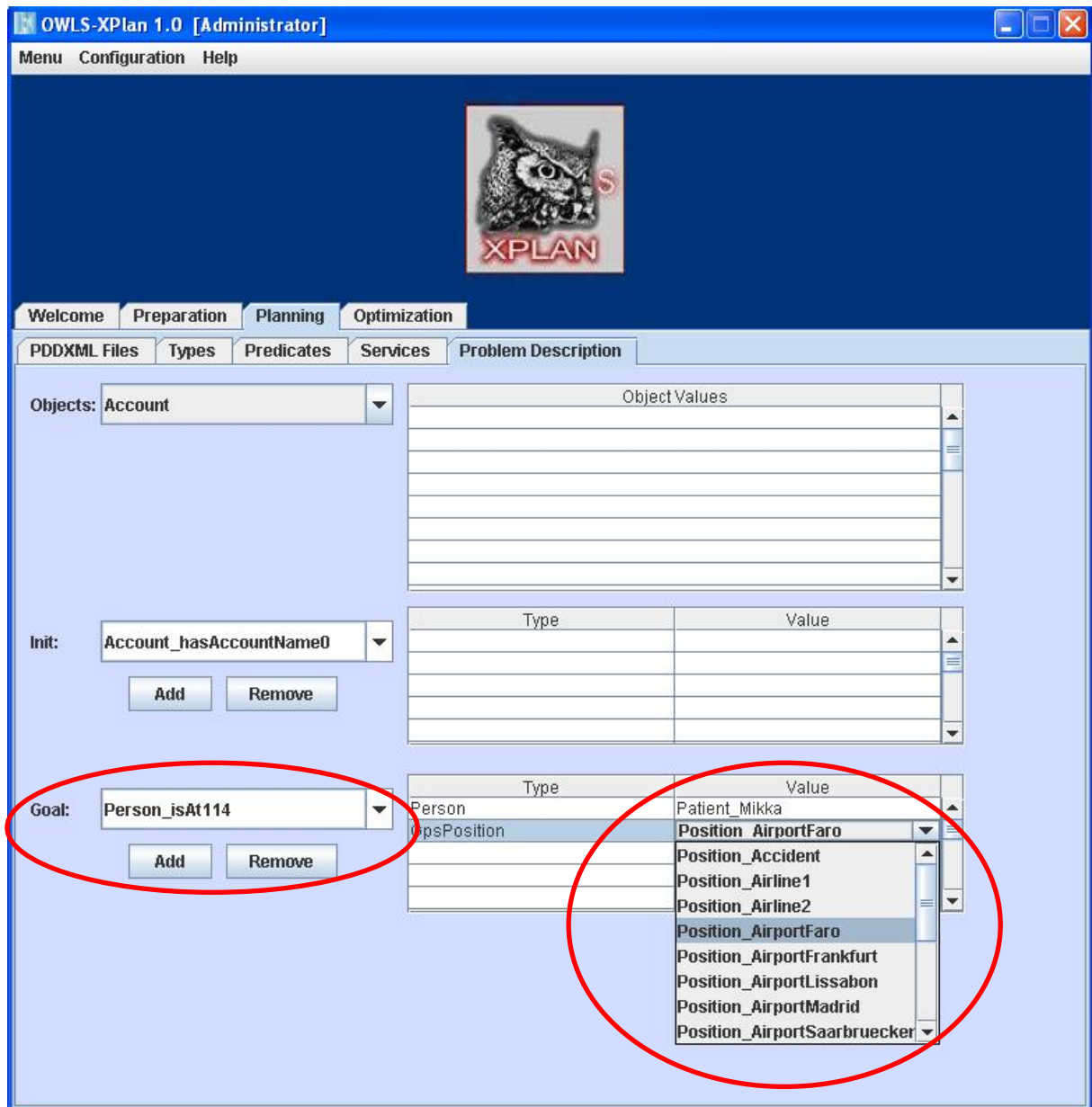


Fig.16: Problem Description tab with modified Goal state table value

In order to see the effect of this change on the composition plan, you must invoke the OWLS-XPlan once again. Therefore you have to click on the *PDDXML Files* tab and then press the *Plan* button. A new composition plan will be generated. Upon successful execution of the OWLS-XPlan, a message box will appear. The window changes automatically to the Optimisation Tab and displays the modified composition plan, see below in (Fig. 17).

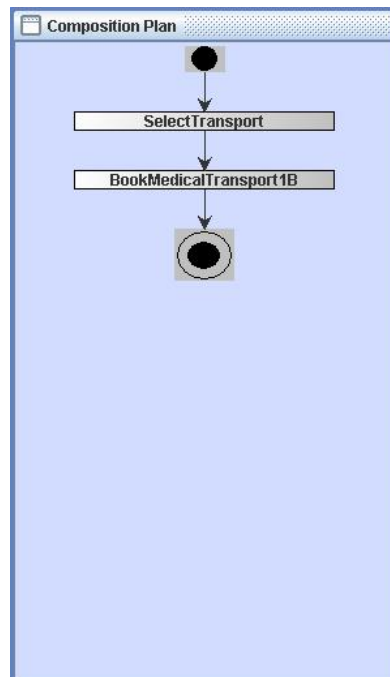


Fig.17: Example Service Composition Plan (modified)

The total number of services in the original composition (see Fig. 14) has changed from 12 to 2 steps (see Fig. 17).

#### ***4.4 QoS Driven Optimization of Composition Plans***

OWLS-XPlan allows doing a post optimisation of the composed plan with respect to a set of given quality of service (QoS) parameters. For this purpose, OWLS-XPlan exploits given knowledge about (semantically) equivalent services and their QoS values. To start an optimisation click on the Optimization Tab (see Fig 18).

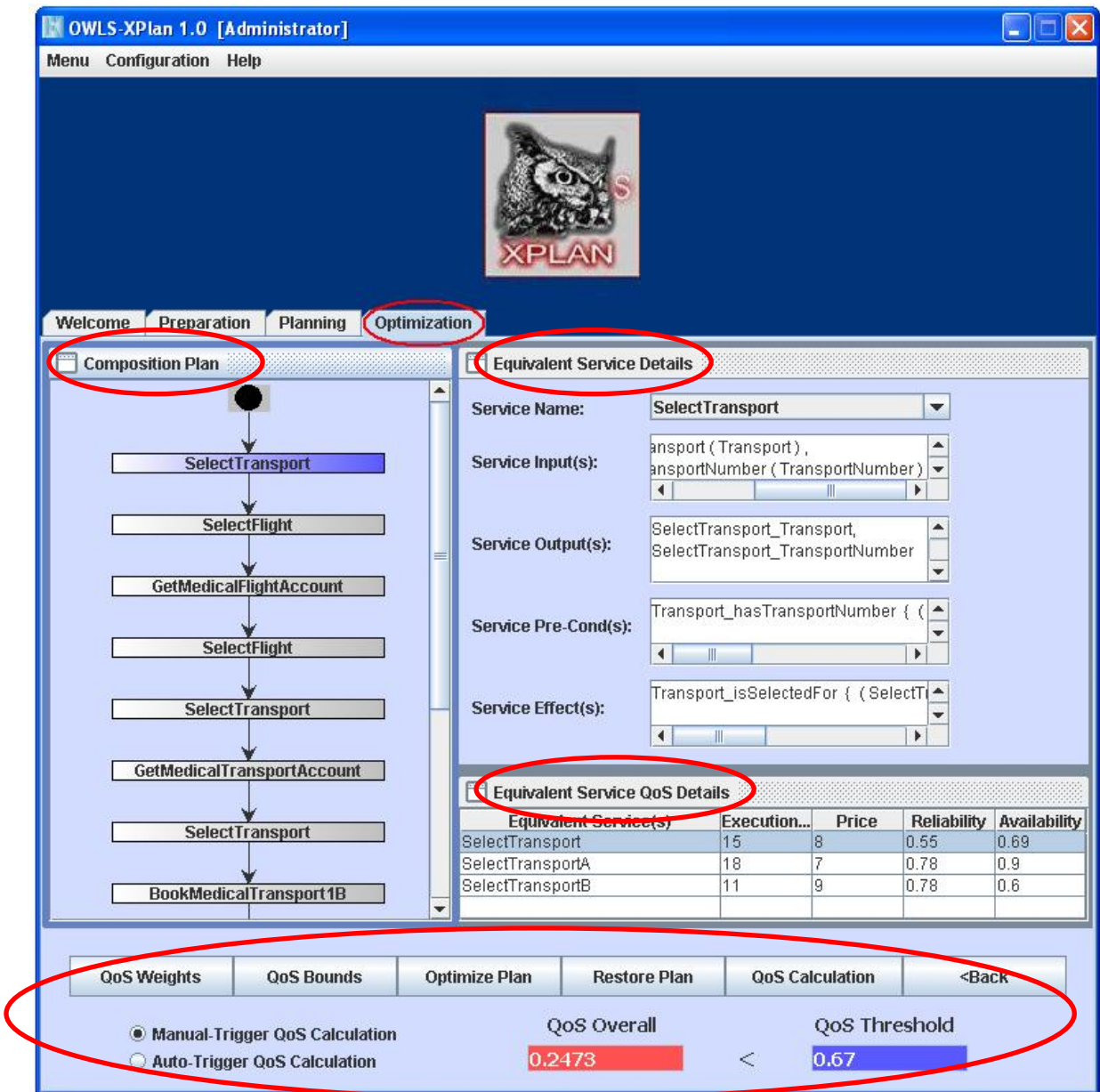


Fig.18: Optimization tab

The *Optimization tab* (Fig. 18) comprises four different sub-panels

- **Composition Plan**
- **Equivalent Service Details**
- **Equivalent Service QoS Details**
- **Control Region**

Each of the above named sub-panels are mutually related to each other and support you to optimize the generated composition plan based on the available set of (semantically) equivalent services' QoS attribute values.

### **Composition Plan**

In the upper left corner of *Optimization tab* (Fig. 18), is shown a sequential *Composition Plan* graph, generated by the OWLS-XPlan. Each node in the graph represents an OWLS service. By double clicking on a service node, the service details are shown in the corresponding *Equivalent Service Details* sub-panel (see Fig. 19).

Fig.19: Equivalent Service Details

Additionally the service's QoS details are shown in lower right *Equivalent Service QoS Details* sub-panel (Fig. 20).

| Equivalent Service(s) | Execution... | Price | Reliability | Availability |
|-----------------------|--------------|-------|-------------|--------------|
| SelectTransport       | 15           | 8     | 0.55        | 0.69         |
| SelectTransportA      | 18           | 7     | 0.78        | 0.9          |
| SelectTransportB      | 11           | 9     | 0.78        | 0.6          |

Fig.20: Equivalent Service QoS Details

### **Equivalent Service Details**

As you can see from (Fig. 19) that *Equivalent Service Details* sub-panel has five different service description fields. **Service Name** is a combo-box containing a list of (semantically) equivalent service names associated with it. By this window the user can manually change any original composition plan's service with any of its (semantically) equivalent service(s) available in the respective Service Name combo box. The (semantically) equivalent service are *Equivalent Service QoS Details* sub-panel (Fig. 20), or the same shown in the lower right corner of the *Optimization tab* (Fig. 18).

### **Equivalent Service QoS Details**

The *Equivalent Service QoS Details* sub-panel (Fig. 20) shows all parameters/QoS attributes that are relevant to perform the OWLS-XPlan QoS optimisation. For example, the lower the value of the service **Execution Duration** and **Price** is, the better it is for the overall QoS of the considered plan. The same holds for **Reliability** and **Availability** attribute values (*probability distribution*).

An example of how to encode the list of equivalent services and their QoS attribute values is given in the file named **Plan\_Equivalent\_Services\_QoS.txt** (included in the *OWLS-XPlan\_1.0 distribution*). The format of entries in this file is simple:

```
[ (SelectTransport) {15:8:0.55:0.69} ||
  (SelectTransportA) {18:7:0.78:0.9} ||
  (SelectTransportB) {11:9:0.78:0.6} ],
```

...

In this example entry, the service *SelectTransport* has two semantically equivalent services *SelectTransportA* and *SelectTransportB*. Given the fixed ordering of the QoS attributes as shown in Fig 20 from left to right, the QoS values are provided in {...} each separated by ':'. For example, service **SelectTransport** has an **Execution Duration** of **15** seconds, **Price** **8 €**, **Reliability** **0.55**, and **Availability** of **0.69**. Similarly, for the next equivalent service in the set, we need to use a separator ||, and the rest goes the same. The [...] brackets are used to separate one set of equivalent services from another.

The other fields (**Service Input(s)**, **Service Output(s)**, **Service Pre-Cond(s)** and **Service Effect(s)**) in the *Equivalent Service Details* sub-panel (Fig. 19) represent the OWLS service's profile model.

Each time when you double click on any of the graph's service-node from the *Composition Plan* sub-panel, the corresponding *Equivalent Service Details* (Fig. 19) and *Equivalent Service QoS Details* (Fig. 20) sub-panels are updated accordingly.

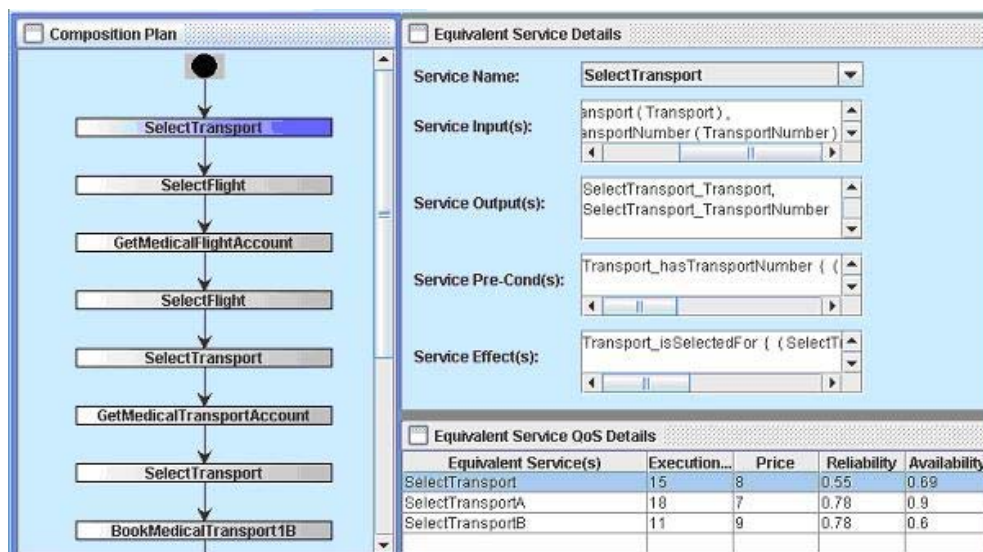


Fig.21: Composition Plan, Equivalent Service Details and Equivalent Service QoS Details

## Control Region

Now we come to the *Control Region* (lower) sub-panel (Fig. 22) of the above shown *Optimization tab* screen (Fig. 18). This sub-panel controls the originally generated or manually modified Composition Plan in a variety of ways.

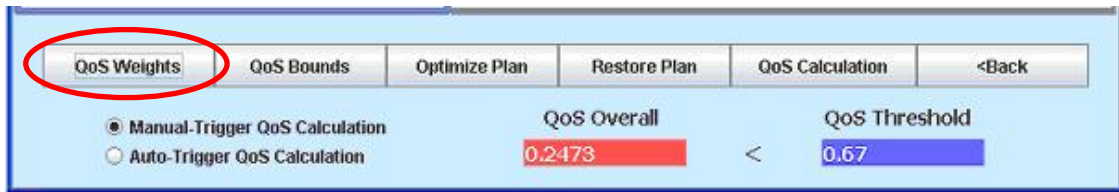


Fig.22: Control Region

First, we'll describe the following screen (Fig. 23), which opens once you click on the **QoS Weights** button, in the Control Region (Fig. 22).

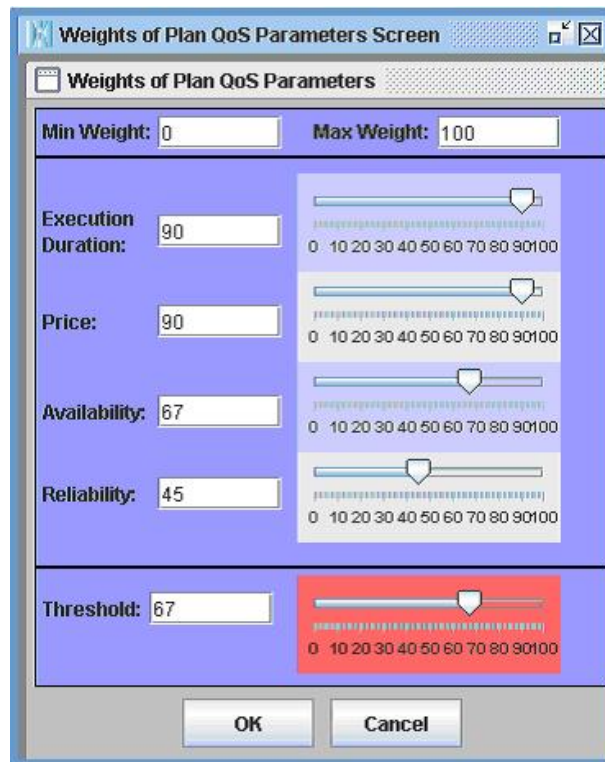


Fig.23: Adjusting the QoS attribute weights used by the optimisation algorithm

Here the user can adjust the weights of each QoS attribute in relation to the others. Figure 23 provides fields to define the **minimum** and **maximum** range related to all attributes. By **Threshold** the user can define an overall QoS value for plans, which are acceptable and do not have to be optimised anymore. The plans should then at least surpass the given threshold.

Now we come to the *QoS Bounds* (Fig. 24) screen, which opens once you click the second functionality button **QoS Bounds** (see Fig. 22) in the Control Region sub-panel.

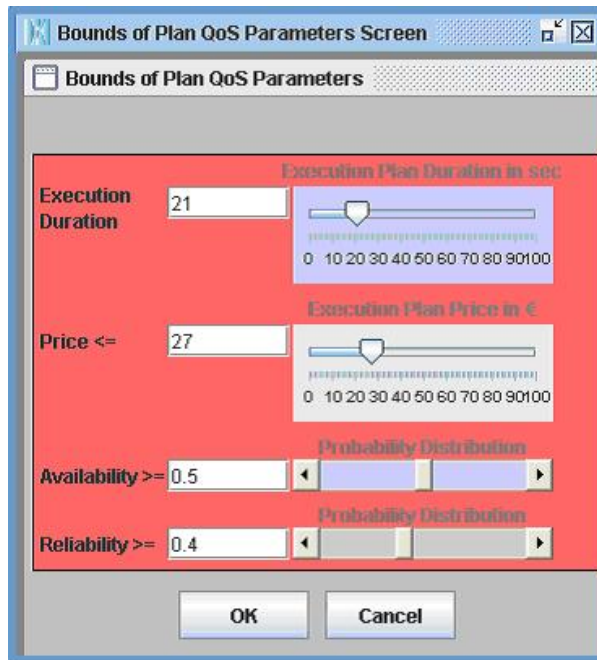


Fig.24: Setting up the bound of each QoS parameter

Figure 24 enables the user to define the global QoS attribute value constraints for a given composition plan. For example, you might want to specify that the overall **Execution Duration** of the plan (its execution run time) should not exceed 21 seconds. Similarly, you can specify the range for values of the QoS attributes **Price**, **Availability** and **Reliability**.

The third button **Optimize Plan** in the *Control Region* (see Fig. 22) is used to start the optimisation process while executing the plan, instead of manually changing the composition plan's services and calculating the overall plan's QoS value (Fig. 25). During optimisation a new composition plan is built, the so called **QoS optimal execution plan**, based on the given list of (semantically) equivalent services and their QoS attribute values (*available for each task*). The execution plan is optimal with respect to the overall QoS value satisfying all given global QoS constraints and not exceeding the given QoS threshold, if possible.



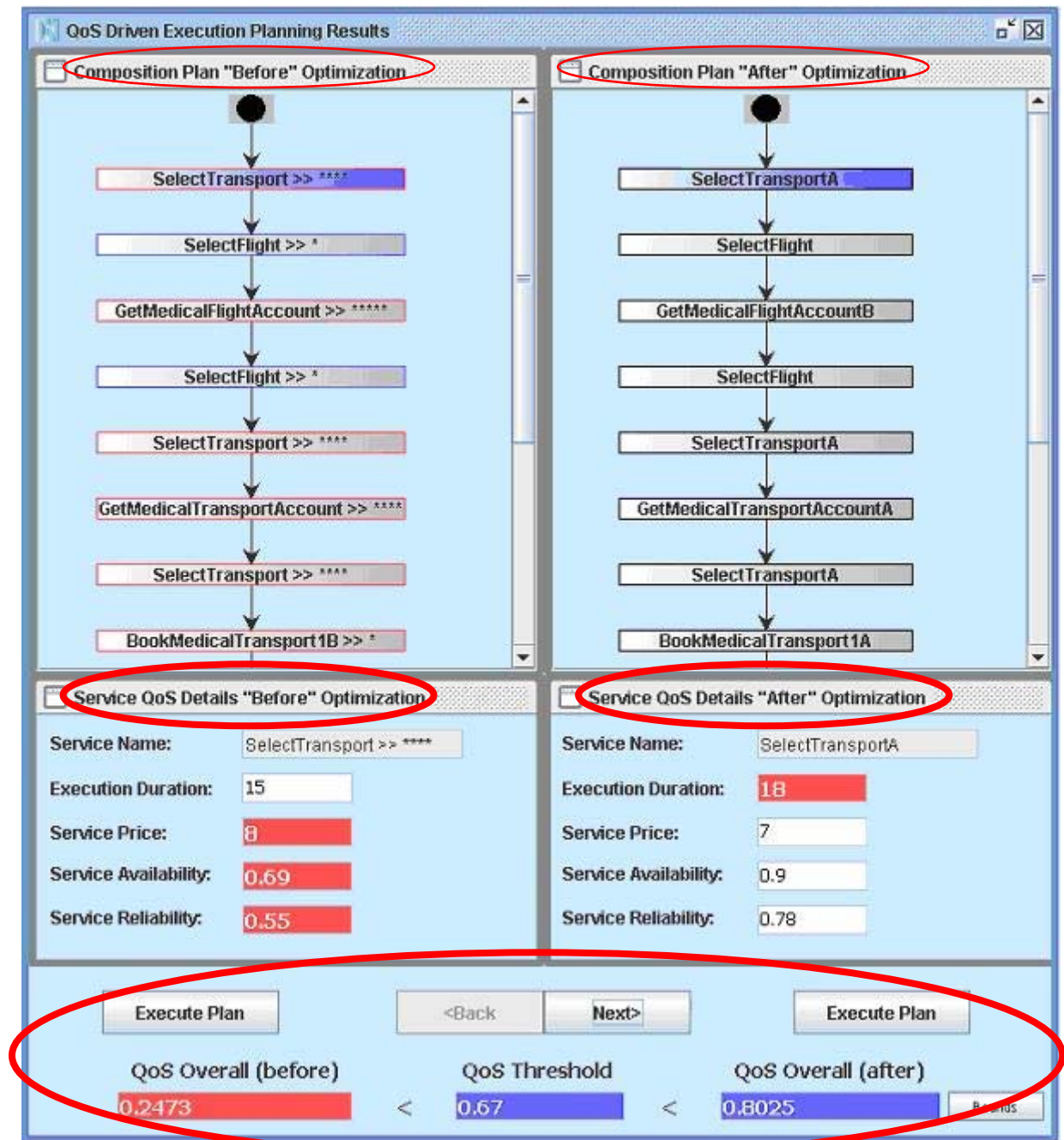


Fig.25: QoS Driven Execution Planning Results

For optimising the plan OWLS-XPlan uses a MILP (*Mixed Integer Linear Programming*) based algorithm. After the optimisation process has been finished a new window opens, as shown in Figure 25. It can be divided into three main sections.

- **Composition Plan (Before/After) Optimization**
- **Service QoS Details (Before/After)**
- **Control Region (Optimization)**

-

## Composition Plan (Before/After) Optimization

The upper section of the QoS Driven Execution Planning Results screen (Fig. 26) represents the graphical view of both, the *Composition Plan “Before” Optimization* and *Composition Plan “After” Optimization* to the user. In this section, the graph shown on the left is the same as the one shown in *Optimization tab* (Fig. 18) at the time when the **Optimize Plan** button was pressed. The graph generated and shown in the right half of the figure 26 is a result of the QoS driven execution planner’s computation. That is, this graph represents an semantically equivalent composition plan to the original one, shown in the left half of figure 26. Hence, it might contain some differently named but equivalent services that are assumed to produce the same results as the original ones but with an overall *optimal* QoS value for the composition plan.

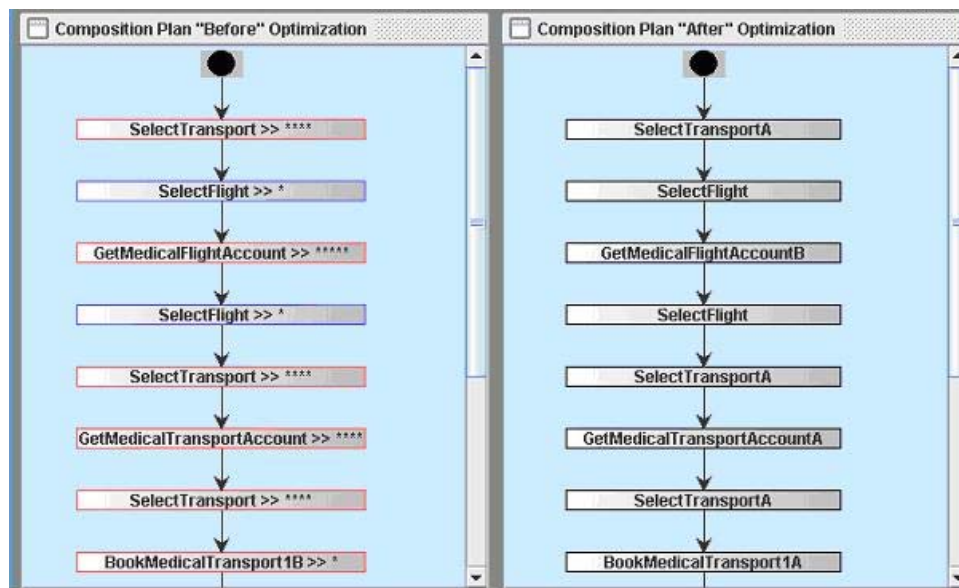


Fig.26: Composition Plan “Before” and “After”

Figure 26 shows coloured boundaries (**red/blue**) of the service-name nodes of the upper left graph (*Composition Plan “Before” Optimization*). The meaning of the color of such a boundary is as follows:

- If it is **red**, then the service has been replaced by its (semantically) equivalent service (*Composition Plan “After” Optimization*).
- If it is **blue**, then the optimised graph (*Composition Plan “After” Optimization*) uses the same service for this particular task.

## Service QoS Details (Before/After)

On pressing the **Next>** button (Fig. 25) for the first time, the first node in both graphs will be highlighted in **blue** and their QoS attributes are displayed in the *Optimization tab* (Fig. 18), as shown in Fig. 27. Each time when the Next> button is pressed; the focus is transferred to the next service node. Similarly, the **<Back** button is used to go back to the previously traversed node.



Fig.27: Service QoS Details “Before” and “After”

Every QoS attribute field in the *Service QoS Detail* section can have three colours - *red*, *blue* and *white*. If the attribute value field has *red* colour means that the corresponding attribute value field, in the right *Service QoS Details “After” Optimization* section must have better QoS attribute value and is filled with *white* colour. Similarly, if any QoS attribute value is *blue* in one section, then in the corresponding section the attribute value will also be *blue* and it is an indication of the fact that the service’s QoS attribute values in both the sections are same.

The “\*” shown along the service name represents the service QoS worthiness. For example, if a service node has four \*\*\*\*. This means that for this particular task there could be four (semantically) equivalent services available and this particular selected service is the worst amongst its equivalent set of services with respect to given QoS attribute values. Similarly, a service node with one \*, means that this could be the best possible service available with respect to given QoS attribute values.

### Control Region (Optimization)

Now looking at the lower section (*Control Region (optimization)*) of the *QoS Driven Execution Planning Results* screen (Fig. 28). The two (<Back and Next>) buttons have the same functionality as the similar buttons in Fig. 27. Now the two **Execute Plan** buttons provide no functionality at the moment. These buttons concern to future work on the OWLS Service Execution Engine. By pressing the **Execute Plan** button either of the composition plan graphs i.e. *Composition Plan “Before” Optimization* or *Composition Plan “After” Optimization* will cause the system to execute the plan.

The controls in Fig. 28 are grouped into three fields, namely **QoS Overall (before)**, which represents the overall QoS value calculated by the QoS driven execution planner for the *Composition Plan “Before” Optimization* graph. Similarly, the **QoS Threshold** field contains the threshold value set by you earlier in the *QoS Weight* screen (Fig. 23). The rightmost field **QoS Overall (after)** is same as the one described above, but contains the overall QoS value for the *Composition Plan “After” Optimization* graph.

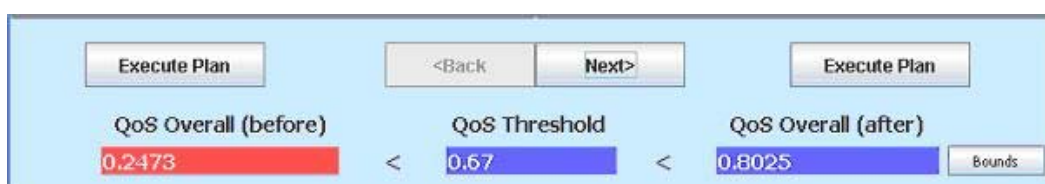


Fig.28: Control Region (Optimization)

The **Bounds** button (on the right lower side) opens a screen (Fig. 29) when pressed, which shows the current *Optimized Plan QoS Bounds*' values achieved during the optimisation.



Fig.29: Optimized Plan QoS Bounds



Fig.30: Control Region

Back to the *Optimization tab* (Fig. 18), the **Restore Plan** button is the fourth button inside the *Control Region* (Fig. 30) sub-panel. As its name suggests, this button is used to restore the manually changed plan to its original form that is the one generated by OWLS-XPlan in the beginning. Once the user pressed this button, a warning dialogue appears (Fig. 31), which asks the user, “**Are you sure, you like to Restore the original Plan**”. If the **OK** button is pressed, the original plan is restored, otherwise on pressing the **Cancel** button, the existing plan remains the same.



Fig.31: Restore Plan dialog

By pressing the **QoS Calculation** button in Figure 30, the QoS value for the existing (*manually changed*) composition plan (*the plan shown in the Composition Plan sub-panel at that time*), will be calculated by the QoS driven execution planner (*a module inside OWL-XPLAN*). The calculated QoS value will be made visible to the user inside the **QoS Overall** text, shown at the bottom of the *Control Region* (Fig. 30). Just adjacent to this field is **QoS Threshold** field, which contains the value set by the user earlier in the *QoS Weights* screen (Fig. 23). If the colour of the **QoS Overall** field is red, this indicates that the overall calculated QoS value of the existing composition plan (*shown in the Composition Plan sub-panel at that time*) is not

satisfactory that is below the threshold mark. Or in other words, if user has performed any manual change(s) in the composition plan, then those change(s) are still not containing appropriate set of services, which should lead to an overall QoS value surpassing the given threshold. Whereas, if the colour of **QoS Overall** field is blue, then we are sure that the **QoS Overall** value must be at least greater than or equal to the threshold and the composition plan may not require automatic QoS driven optimization.

Now in the top rightmost side of the *Control Region* (Fig. 30) is the **<Back** button, which, if pressed, just switches the tab from *Optimization* to *Planning*.

In the left bottom of the *Control Region* (Fig. 30), we have one radio-button group containing two radio-buttons, namely **Manual Trigger QoS Calculation** and **Auto Trigger QoS Calculation**. By default “Manual Trigger QoS Calculation” is checked, which means that each time you make any changes in the composition plan by replacing any service with its (semantically) equivalent service, the QoS driven execution planner will not automatically be invoked, but perform its task including this change only when you press the **QoS Calculation** button. If you set the radio-button to automatically trigger the QoS Calculation on each service change, the QoS driven execution planner would be invoked and calculate the overall QoS value.

## 5 Disclaimer

OWLS-XPlan is free for non-commercial use only. You can redistribute it and/or modify it under the terms of the Mozilla Public License version 1.1 as published by the Mozilla Foundation at <http://www.mozilla.org/MPL/MPL-1.1.txt>

Copyright 2006, DFKI GmbH, Germany

Developed by Andreas Gerber, Marcus Schmidt, Suleman Butt and Matthias Klusch

## 6 Support and contact

Please report any bugs and errors you encounter to us; we are continuously working to improve OWLS-XPlan!

Errors can be reported by emailing the developers (see below), or by using the online tracker system available at the project homepage at [semwebcentral.org](http://semwebcentral.org)

For OWLS-XPlan software support and help, please contact

- Dr. Andreas Gerber at [agerber@dfki.de](mailto:agerber@dfki.de)
- MSc. Suleman Butt at [butt\\_suleman@hotmail.com](mailto:butt_suleman@hotmail.com)
- Dipl.-Inform. Markus Schmidt at [Markus.Schmidt@dialogika.de](mailto:Markus.Schmidt@dialogika.de)

For related general and scientific questions, please contact:

- Dr. Matthias Klusch at [klusch@dfki.de](mailto:klusch@dfki.de), <http://www.dfki.de/~klusch>