

Inspect, Edit and Debug PDDL Documents: Simply and Efficiently with PDDL Studio

Video presentation

Tomas Plch¹ Miroslav Chomut² Cyril Brom³ Roman Barták⁴

Faculty of Mathematics and Physics, Charles University in Prague

¹tomas.plch@gmail.com ²chmirko@gmail.com ³brom@ksvi.mff.cuni.cz ⁴bartak@ktiml.mff.cuni.cz

Introduction

In this short description, we describe our video presentation of our PDDL Studio project. The presentation is split into five sections 1) basic layout and controls (top menu, auxiliary menus, window resizing etc.), 2) project management, 3) main features – bracket matching, error detection etc., 4) XML export and import, and 5) planner integration and execution. We describe the various parts of the application in detail.

Basic Layout

This section is presented in the *PDDLStudio_1.avi* which is split into 3 subsection 1) application layout (0:00 – 0:10), 2) main menu (0:40 – 1:10) and auxiliary menu (0:18 – 0:35), 3) window resizing (1:10 – 1:52). The application layout is presented in Figure 1.

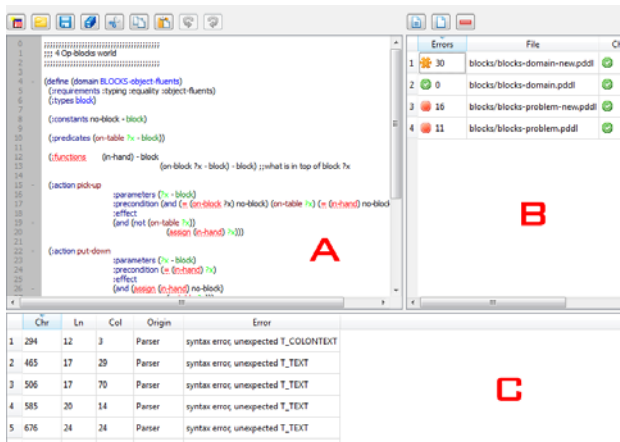


Figure 1: PDDL Editor window with Editing Window (A) having highlighted syntax, Interactive Error Report (C) and Project Manager (B) with additional information about file status (number of errors, save status etc.)

The main menu (Figure 2 (A)) contains all necessary functionality and is located on the top left of the application window. Auxiliary menus are located above their respective areas – above the editor window (Figure 1 (A)) and the project management window (Figure 1 (B)). The editor auxiliary menu (Figure 2(B)) displays most of the used functionalities for file editing (e.g. load, save, copy, paste, undo etc.). The project management auxiliary menu (Figure 2(C)) displays functionality for project management (e.g. adding new or existing files, removing files from the project).

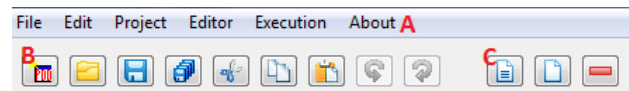


Figure 2: Main menu (A), Editor auxiliary menu (B) and Project manager auxiliary menu (C)

Project management

This section describes the project management shown in *PDDLStudio_2.avi*. It is split into 5 subsections 1) creating of a new project (0:00 – 0:18), 2) adding new PDDL document to the project, editing and saving the content (0:18 – 0:40), 3) closing the project (0:40 – 0:50), 4) opening a project (0:50 – 2:00), 5) removing and adding files to the project (2:00 – 2:25).

Project operations

To create, save and close a project, the user has to utilize the appropriate icon in the editor's auxiliary menu or in the main menu. The project is saved into a file with the extension *pddl_proj*.

File operations in respect to the project are performed by the project manager's auxiliary menu (Figure 2(C)) – adding new file, adding existing file, removing file.

PDDL document operations

PDDL documents, when added to a project, can be edited via the Editor window (Figure 1(A)). The status of a file can be seen in the Project manager window (Figure 1(B)). Every file in the project is listed in a table containing the following information (Figure 3): 1) file number – used in execution scripts etc., 2) error listing with symbols accompanied with error count (the presently selected file has a *star* symbol instead of the green or red symbol), 3) file path relative to the project's location, 4) change indicator.

	Errors	File	Change
1	30	blocks/blocks-domain-new.pddl	
2	0	blocks/blocks-domain.pddl	
3	16	blocks/blocks-problem-new.pddl	
4	11	blocks/blocks-problem.pddl	

Figure 3: Project manager's file info table

Main features

The main features of PDDL Studio are shown in *PDDLStudio_3.avi* and are as follows: 1) error detection and localization (0:00 – 0:30), 2) hints and code completion (0:30 – 1:00), 3) syntax highlighting, 4) code collapsing (1:00 – 1:30), 5) common editor features (e.g. font and color changes, line numbering) (1:30 – 2:05).

Error detection and localization

If a PDDL document contains errors, this is listed in the error table within the error console (Figure 1(C)). The error console is interactive – if the user picks an error from the list and clicks on it, the editor jumps to error in question. The error table (Figure 4) contains the following columns – 1) error number, 2) location indicated by line, character and column, 3) origin of the error – where the error occurred (e.g. Parser errors, invalid formula parameters etc.), 4) error description.

	Chr	Ln	Col	Origin	Error
1	294	12	3	Parser	syntax error, unexpected T_COLONTEXT
2	465	17	29	Parser	syntax error, unexpected T_TEXT
3	506	17	70	Parser	syntax error, unexpected T_TEXT
4	585	20	14	Parser	syntax error, unexpected T_TEXT
5	676	24	24	Parser	syntax error, unexpected T_TEXT
6	723	26	20	Parser	syntax error, unexpected T_TEXT
7	853	31	29	Parser	syntax error, unexpected T_TEXT
8	870	31	46	Parser	syntax error, unexpected T_TEXT

Figure 4: Error report table

The errors are also marked within the editor's text by an underline of the actual error detected (Figure 5).

```
:parameters (?x - block ?y - block)
:precondition (and (= (in-hand ?x) (= (on-block ?y) no-block))
:effect
(and (assign (in-hand) no-block)
      (assign (on-block ?y) ?x)))
```

Figure 5: Underline of errors in PDDL documents

Hints and code completion

Hints and code completion are realized by a box appearing near the editor cursor when the user is typing. If the user continues typing, the system adapts and filters the available choices (Figure 6). These hints are based on the current context of the edited piece of code. The user can navigate the box with arrow keys and the selection is made by pressing the enter key.

```
:parameters (?x - block ?xanother - blockAnother)
:precondition (and (clear ?x) (ontable ?x) (handempty) ())
:effect
(and (not (ontable ?x))
      (not (clear ?x))
      (not (handempty))
      (holding ?x)))
```

clear

handempty

holding

on

ontable

Figure 6: Code completion and hints

Syntax Highlighting

Syntax highlighting is based on the language elements of PDDL (Figure 5). The distinguished elements are: 1) names, 2) variables, 3) errors underscore, 4) language keywords, 5) predicates, 6) types, 7) highlighted brackets at the cursor's position and bracket's counterpart, and 8) normal text. All these colors can be changed via the editor's interface.

Code collapsing

Code collapsing is realized on the highest level of language elements to maintain readability (Figure 7). It is facilitated by the '-' sign and '+' sign (expansion) located within the line numbering column of the editor. The collapsed portions of code are maintained until the file or project is closed.

```
14 + (:action pick-up :| 14 - (:action pick-up :|
20 - (:action put-down 15 - (:action pick-up :|
21 - (:action put-down 16 - (:action pick-up :|
22 - (:action put-down 17 - (:action pick-up :|
23 - (:action put-down 18 - (:action pick-up :|
24 - (:action put-down 19 - (:action pick-up :|
25 - (:action put-down 20 - (:action pick-up :|
26 - (:action put-down 21 - (:action put-down
27 - (:action stack 22 - (:action pick-up :|
28 - (:action stack 23 - (:action pick-up :|
29 - (:action stack 24 - (:action pick-up :|
30 - (:action stack 25 - (:action pick-up :|
```

Figure 7: Code collapsing of action element

Other editor features

The editor also provides commonly present features seen in other editing tools – line numbering and font changes. The font changes are realized over a standardized interface. The line numbering is present on the left side of the editing window.

XML Export/Import

PDDL Studio also provides the capability to export and import our PDDL XML format. This is shown in the *PDDLStudio_4.avi* which contains the following sections: 1) exporting XML to PDDL (0:00 – 1:15), 2) importing XML into PDDL Studio (0:15 – 2:20).

Planner integration and execution

PDDL Studio is capable to make use of an existing planner that can be executed from a command line prompt. This is presented in the *PDDLStudio_5.avi* where we used the BlackBox planner on our PDDL project. We use a simple scripting language, where various information from the PDDL Studio (e.g. file names, file numbering, etc.) can be used within the command parameters of the executed planner.

Availability

The project can be downloaded at <http://amis.mff.cuni.cz/PDDLStudio/download.html> and the video files can be accessed at <http://amis.mff.cuni.cz/PDDLStudio/video.html>.