Properties Panel 1.1 Requirements Specification

1. Scope

1.1 Overview

The properties panel component provides a number of different SWING panels that allow the user to set different properties of model elements and groups of elements. It also provides a way to signal the listeners of changes.

This new version of the component will update the layouts for each individual panel to match prototype images given. The new layouts take up less space and are organized better for the tool. Almost none of the current functionality of the panels will change, only their layouts.

1.2 Logic Requirements

1.2.1 Reorganization

Prototype images have been supplied showing the different possible property panel layouts for the different elements in the tool. Each layout should be updated to match the prototype images shown. This may require changes to the layouts chosen for each panel. In these cases, the developer is free to choose the best layout that will properly match the prototype. The prototype is expected to be matched in this component **exactly** like it is shown in the images provided unless approved changes are made by the PM.

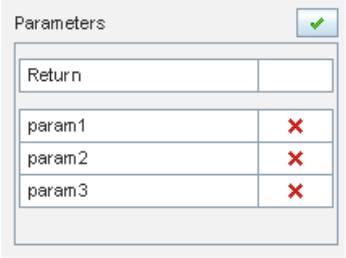
1.2.2 Resizing

When resizing the layouts, the vertical spacing between elements should stay constant. For instance, in the "11-Operations.png" prototype image, if the panel is resized taller, the name, owner, concurrency, modifiers, and visibility areas should all stay grouped together as they are, without adding vertical space between the elements

When resizing horizontally, only the stereotype panel should expand to take up additional space, but it should stop expanding in size when it reaches the size shown in the "06-Lifeline.png" image, and any additional space should be empty. Note that the column in the stereotypes grid that contains the "X" to remove the element should be constant size, the same size seen for the parameters grid in "11-Operations.png". There is a bug in the prototype where the column with the "X" is shown taking up half the size of the grid.

1.2.3 Parameters grid

The only functionality changes affect how the "Parameters" grid works, as seen in "11- Operations.png".



Note that the "Return" value is separated from the rest of the elements in the grid. This is needed to differentiate the return value from the rest of the parameters. In the prototype, no "X" is shown next to the "Return" value, but one is needed. The only difference is that when clicking this "X", instead of immediately deleting the return value, a confirmation dialog is displayed that states "Are you sure you want to delete the return value for this operation?"

1.3 Required Algorithms

None

1.4 Example of the Software Usage

This component will be used in the TopCoder UML Tool so the user can change properties of different elements.

1.5 Future Component Direction

None

2. Interface Requirements

2.1.1 Graphical User Interface Requirements

The updated layouts must match what is shown on the prototype images supplied

2.1.2 External Interfaces

None

2.1.3 Environment Requirements

- Development language: Java 1.5
- Compile target: Java 1.5

2.1.4 Namespace

com.topcoder.gui.panels.properties

3. Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?

- None

3.2 Technical Constraints

3.2.1 Are there particular frameworks or standards that are required?

None.

3.2.2 TopCoder Software Component Dependencies:

- Configuration Manager 2.1.5
- UML Model components

**Please review the TopCoder Software component catalog for existing components that can be used in the design.

3.2.3Third Party Component, Library, or Product Dependencies:

None

3.2.4QA Environment:

- Solaris 7
- RedHat Linux 7.1
- Windows 2000
- Windows 2003

3.3Design Constraints

The component design and development solutions must adhere to the guidelines as outlined in the TopCoder Software Component Guidelines. Modifications to these guidelines for this component should be detailed below.

3.4Required Documentation

3.4.1Design Documentation

- Use-Case Diagram
- Class Diagram
- Sequence Diagram
- Component Specification

3.4.2Help / User Documentation

XML documentation must provide sufficient information regarding component design and usage.