

Document Tree 1.1 Requirements Specification

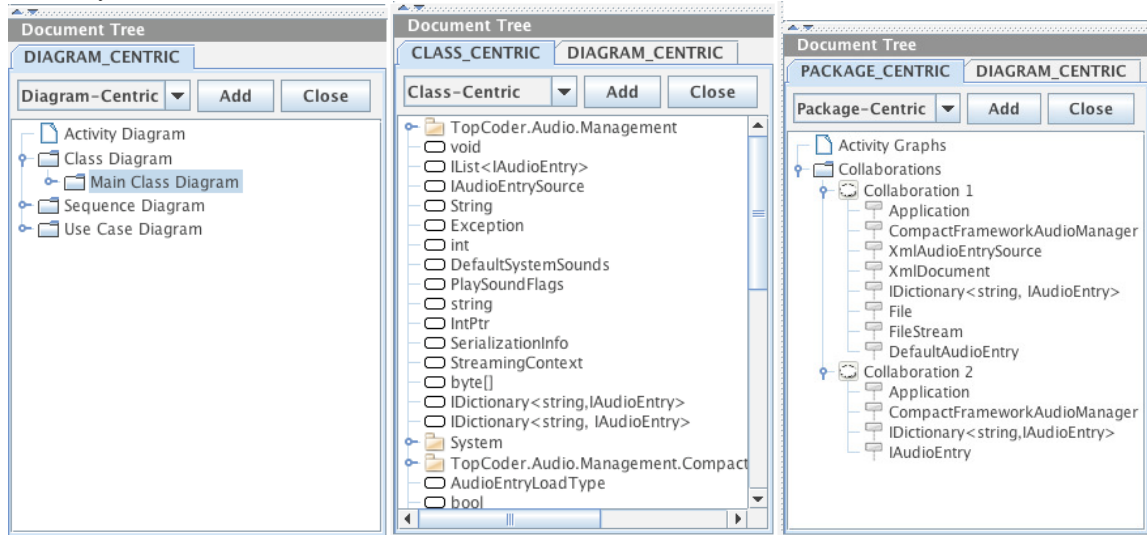
1. Scope

1.1 Overview

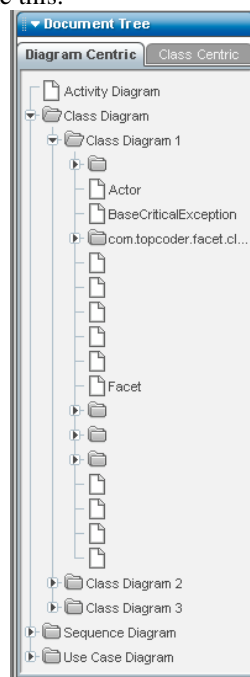
The document tree currently is displayed in the bottom left corner of the UML Tool, and it provides a visual display of the internal model of elements in a given project. The elements can be clicked on to display them in the diagram view, and can also be renamed through the document tree. This development component will update the overall layout of the document tree to match updates to the overall application.

1.2 Logic Requirements

Currently the document tree looks like this, with each of its different tabs:



The tree is going to be reorganized to look like this:





Note that for this update, the look and feel of the tree **do not** need to be updated. This is going to be handled separately for the application as a whole. We are only worried about the layout of the tree.

1.2.1 Control removal

In the diagram tree currently, there is a drop-down to choose new tabs to add to the tree view, as well as “Add” and “Close” buttons. The drop-down, “Add”, and “Close” buttons should be removed.

1.2.2 Static tabs

In the new diagram tree layout, only two tabs are open by default, “Diagram Centric” and “Class Centric”.

1.2.3 Configurability

The tabs shown in the view should be configurable, so even though “Diagram Centric” and “Class Centric” are the defaults, we should be able to change the tabs and order of the tabs shown without having to change code in the tree. The new configuration parameters should be added and documented properly.

1.3 Required Algorithms

None

1.4 Example of the Software Usage

This component will be used in the TopCoder UML Tool as the document tree of the GUI.

1.5 Future Component Direction

None

2. Interface Requirements

2.1.1 Graphical User Interface Requirements

The updated layout 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 1.6

2.1.4 Namespace

com.topcoder.gui.trees.document

3. Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?

- The tree icons
- The tabs and the order they are shown

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
- UML Model Manager 1.0

**Please review the [TopCoder Software component catalog](#) for existing components that can be used in the design.



3.2.3 Third Party Component, Library, or Product Dependencies:

None

3.2.4 QA Environment:

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

3.3 Design 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.

3.4 Required Documentation

3.4.1 Design Documentation

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

3.4.2 Help / User Documentation

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