XMI Writer UML Model Plugin 1.0 Requirements Specification

1. Scope

1.1 Overview

The XMI Writer UML Model Plugin component provides the ability to write the Model structure to XMI format. The transformer will be invoke by the XMI Writer with the root Model element. It will write the response to the output print stream.

1.2 Logic Requirements

1.2.1 Model2XMITransformer

The component will provide this class. The transformers will receive the model:Model as an object and the output print stream. It will transform the model to XMI and write the response to the output stream.

The elements should be written completely only if they are present on the logical containment hierarchy. Otherwise, a reference should be written.

1.2.2 Unknown element type

There should be an option to ignore the unknown elements, or to throw an exception.

1.2.3 DTD file

The XMI output will conform to the provided DTD schema.

1.3 Required Algorithms

None.

1.4 Example of the Software Usage

The component will be used in the TopCoder UML Tool to write the XMI files.

1.5 Future Component Direction

None.

2. Interface Requirements

2.1.1 Graphical User Interface Requirements

None.

2.1.2 External Interfaces

The design must follow the interface found in the class diagram with the component interfaces. The designer is encouraged to add to the existing interface, but not to remove anything.

2.1.3 Environment Requirements

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

2.1.4 Package Structure

com.topcoder.uml.xmi.writer.transformers.model

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:

- UML Model components
- UML Model Manager

**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

- Design documents must clearly define intended component usage in the 'Documentation' tab of Poseidon.