Technical Inspection Check List

For GMoDS Visualizer and Test Driver

Submitted in partial fulfillment of the requirements of the degree of MSE

Mike Fraka
CIS 895 – MSE Project
Kansas State University

Table of Contents

1 Introduction

This document provides references to items from the GMoDS Visualizer and Test Driver project to be inspected, the inspection team, and the check list to use during the inspection.

2 References

- All items to inspect can be found at http://people.cis.ksu.edu/~mfraka/FrakaMSE.html.
- Sample technical inspection cover letters can be found at http://mse.cis.ksu.edu/deepti/ in the Phase 3 column.

3 Items to Inspect

3.1 System Architecture Design Document 1.0

1. System Architecture (Section 4)
    a. System Components (Section 4.1)
    b. System Component Responsibilities (Section 4.2)
    c. System Interface Specifications (Section 4.3)
    d. System Architecture Design Rationale (Section 4.4)
2. GMoDS Test Driver Architecture (Section 5)
    a. GMoDS Test Driver Decomposition Class Diagram (Section 5.1)
    b. GMoDS Test Driver Module Responsibilities (Section 5.1.1)
    c. GMoDS Test Driver Interface Specifications (Section 5.1.2)
    d. GMoDS Test Driver Design Rationale (Section 5.1.3)
3. GMoDS Architecture (Section 8)
4. USE/OCL Model (Section 9)

4 Inspectors

- Shylaja Chippa
- Kyle Hill

5 Formal Inspection Check List

Table 1 Formal Inspection Check List

Inspection Item	Pass/Fail/Partial	Comments
The system component diagram (Figure 2) uses legal UML elements.		
Section 4.1 clearly explains the elements of the system component diagram.		

Inspection Item	Pass/Fail/Partial	Comments
Table 1 clearly explains the responsibilities of each system component.		
Table 2 clearly specifies the GMoDS Test Driver main program interface.		
Table 3 clearly specifies the GMoDSVisualizer interface.		
Table 4 clearly specifies the TestDriver interface.		
Section 4.4 clearly explains the rationale for the system architecture.		
The GMoDS Test Driver architectural module class diagram (Figure 3) uses legal UML elements.		
Table 5 clearly explains the responsibility of each GMoDS Test Driver architectural class or interface (Note: GoalTree is a GMoDS interface not a GMoDS Test Driver interface).		
Table 6 clearly specifies the GoalEvent interface.		
Table 7 clearly specifies the EventScript interface.		
Section 5.1.3 clearly explains the rationale for the GMoDS Test Driver architecture.		
The GMoDS architectural class diagram (Figure 14) uses legal UML elements.		

Inspection Item	Pass/Fail/Partial	Comments
Section 8 clearly explains the rationale for Figure 14 elements.		
Classes in the USE/OCL model (section 9) are consistent with the classes in Figure 14.		
Attributes in the USE/OCL model (section 9) are consistent with the corresponding classes in Figure 14.		
Associations in the USE/OCL model (section 9) are consistent with associations in Figure 14.		
Multiplicities in the USE/OCL model (section 9) are consistent with multiplicities on the corresponding associations in Figure 14.		