Applying Broadcasting/Multicasting/Secured Communication to agentMom in Multi-Agent Systems

Formal Inspection Checklist

Version 1.0

This document is submitted in partial fulfillment of the requirements for the degree MSE.

Chairoj Mekprasertvit
CIS 895 – MSE Project
Kansas State University
Summer 2003

1. Introduction

The purpose of this document is to provide a formal checklist for the architecture design documents of the project "Applying Broadcast/Multicast/Secured Communication to agentMom in Multi-Agent Systems. Formal technical inspection process will ensure the quality of the software design. Two independent MSE students will perform the inspection and provide the formal report on the result of their inspection.

2. Item to be inspected

Architecture design documents of the project "Applying Broadcasting/Multicasting/ Secured Communication to agentMom in Multi-Agent Systems" including use cases diagram, class diagram and sequence diagram will be inspected.

The following documents will be supplied to each inspector for inspection and references:

1.) Software Requirements Specification version 1.0 *
2.) Project Overview version 1.0 *
3.) agentMom User's Manual *
4.) Class Diagram
5.) Sequence Diagram
6.) Use Case Diagram

Note: The star (*) indicates that the document is available only for references, not for inspection.

3. Organization

Supervisory Committee consisted of:
Dr. Scott A. DeLoach
Dr. David Gustafson
Dr. William Hankley

Major Professor:
Dr. Scott A. DeLoach

Developer:
Chairoj Mekprasertvit

Formal Technical Inspector consisted of
Madhukar Kumar
Acharaporn Pattaravanichanon

4. Formal Technical Inspection Checklist

Item	Pass/Fail/Partial	Comment
1. All the symbol used in the use case diagram conforms to the UML standard.		
2. All the symbol used in class diagram conforms to UML standard.		
3. All the symbol used in Sequence diagram conforms to UML standard		
4. If there is a message passing between objects in sequence diagram, association relationship in class diagram is defined.		
5. Each message in sequence diagram is a method in class diagram.		
6. Use case scenarios and description are clear. Example: use case scenarios are clearly explained.		
7. Class diagram and description are clear. Example: role and responsibility of each class are clearly explained.		
8. Sequence diagram and description are clear.		
9. Names used in class diagram indicated their meaning. Example: class MulticastConversation indicates that it is used for sending and receiving multicast message.		
10. The defined public attributes should be accessible to the outside class.		
11. The defined private attributes should be accessible only within the class.		
12. The defined protected attributes should be accessible by subclass or other classes in the agentMom package.		