**System Architecture Design**

For Multiagent Control of Traffic Signals

Version 1.0

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

Bryan Nehl

CIS 895 – MSE Project

Kansas State University

Table of Contents

[1 Introduction 3](#_Toc319500625)

[2 References 3](#_Toc319500626)

[3 Architecture 3](#_Toc319500627)

[3.1 System Analysis 3](#_Toc319500628)

[3.2 Component Design 5](#_Toc319500629)

[3.3 Component Interface Specification 7](#_Toc319500630)

[3.4 System Design Rationale 7](#_Toc319500631)

[4 Mid-Level Design 7](#_Toc319500632)

[5 Component Interaction 8](#_Toc319500633)

[6 USE/OCL Model 9](#_Toc319500634)

# Introduction

This document provides system design information for the MultiAgent Control of Traffic Signals (MACTS) system. This system is used to simulate agent based control of traffic light signals. This document covers the system components and component interfaces. However, it does not cover all of the interfaces methods in detail. A system analysis diagram as well as a high-level overview of the whole system are included. Mid-Level design is also included for all of the components. A sequence diagram is included which shows how the system components interact during run time.

# References

1. “Vision Document” available at <http://people.cis.ksu.edu/~bnehl/>.

# Architecture

This section documents the system component design, the interfaces of those components and provides high-level design with rationale for design within the system context.

## System Analysis

Clearly describe the high level relationship between model elements

This system context diagram shows how the components of the MACTS system interact with each other and with the external systems.

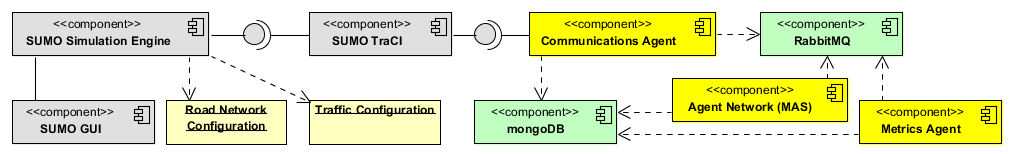


Figure 1System Context Diagram

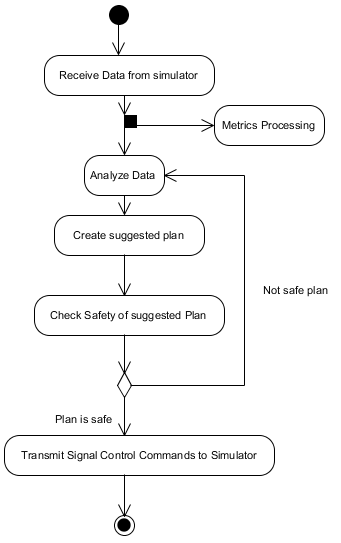


Figure 2Basic Procesing for single simulation step

## Component Design

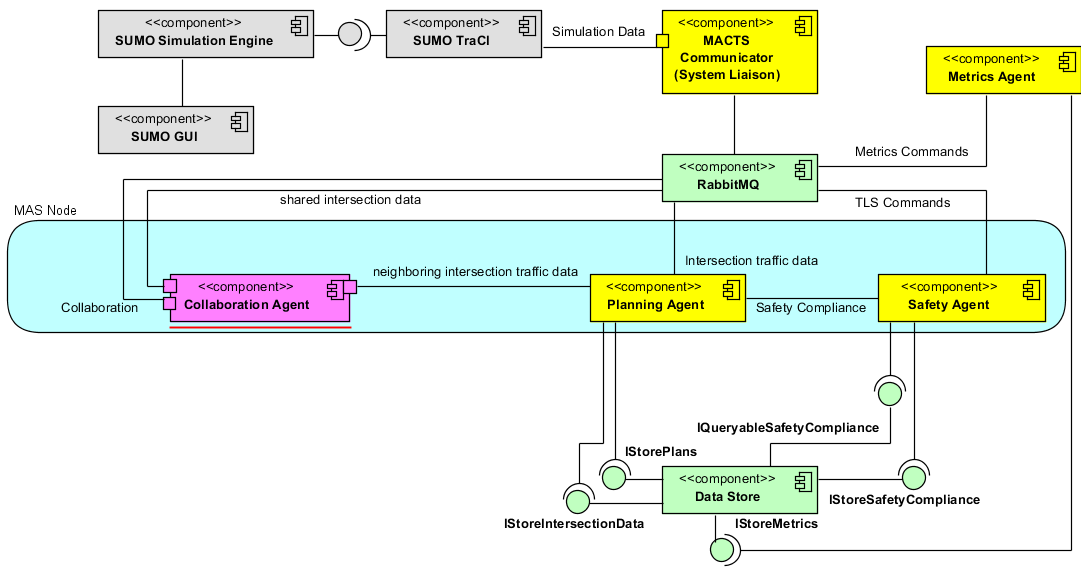


Figure 3MACTS with single MAS Node

*Description of system component responsibilitieshere.*

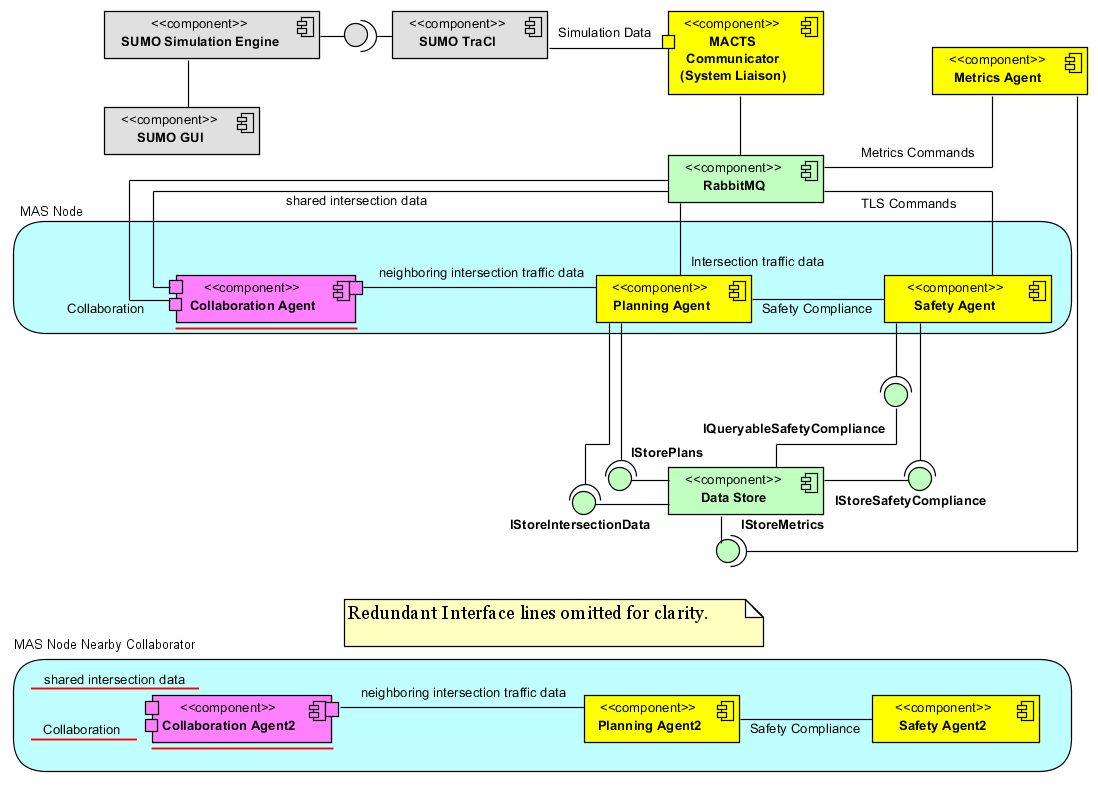


Figure 4MACTS with collaborating MAS Node

*Description of how additional MAS Node interacts here.*

## Component Interface Specification

*Documentation of key interface members for system components here.*

## System Design Rationale

*Description of rationale behind design here.*

# Mid-Level Design

*Class Diagrams here*

# Component Interaction

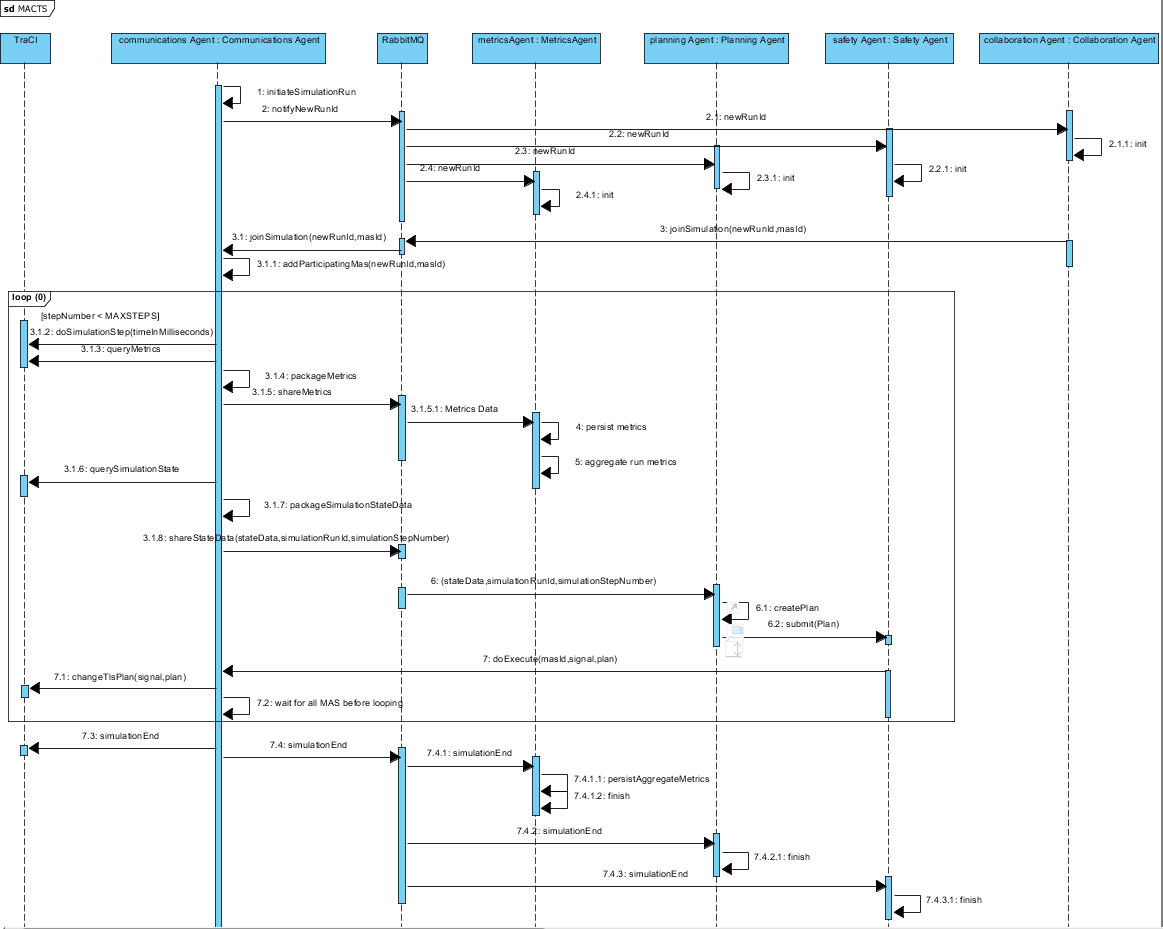


Figure 5Sequence diagram for process interactions

# USE/OCL Model

-- CIS 895 MSE Project Formal Specification MACTS Architecture

--

-- macts.use

--

-- The MACTS model rendered in USE OCL,

--

-- Author: Bryan Nehl

--

model Macts

-- classes -------------------

-- external, only one interface to this

class TraCI

end

-- abstract, no instances of

class Agent

end

-- one

class CommunicationsAgent < Agent

end

-- one

class MetricsAgent < Agent

end

-- multiple uses of

class DataStore

end

-- MAS Node

-- abstract

-- may only have one "PlanningAgent" type

class MasNode

attributes

planningAgent : PlanningAgent

safetyAgent : SafetyAgent

collaborator : CollaborationAgent

end

class PlanningAgent < Agent

end

class StandardTimingBasedAgent < PlanningAgent

end

class ReactiveAgent < PlanningAgent

end

class CollaborativeAgent < PlanningAgent

end

class GeneticAgent < PlanningAgent

end

class SafetyAgent < Agent

end

class CollaborationAgent < Agent

end

-- associations -----------------

association persists between

PlanningAgent[1] role planproducer;

DataStore[\*] role datastore;

end

association interacts between

TraCI[1] role simulator;

CommunicationsAgent[1] role liaison;

end

-- constraints --------------------

constraints

context mn:MasNode

inv planningAgentIsAPlanningAgent:

mn.planningAgent.oclIsKindOf(PlanningAgent)