**Project Evaluation**

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](#_Toc322371432)

[1.1 References 3](#_Toc322371433)

[2 Process 3](#_Toc322371434)

[2.1 Problems Encountered 3](#_Toc322371435)

[2.2 Estimates 3](#_Toc322371436)

[2.2.1 Source Lines of Code 3](#_Toc322371437)

[2.2.2 Rework 3](#_Toc322371438)

[2.2.3 Project Duration 3](#_Toc322371439)

[2.3 Lessons Learned 3](#_Toc322371440)

[3 Products 4](#_Toc322371441)

[3.1 Quality 4](#_Toc322371442)

[3.1.1 Rework Ratio 4](#_Toc322371443)

[3.1.2 Test Coverage 4](#_Toc322371444)

[3.1.3 Cyclomatic Complexity 4](#_Toc322371445)

[3.1.4 MTBF 4](#_Toc322371446)

[3.2 Future Work 4](#_Toc322371447)

[3.2.1 More Generic 4](#_Toc322371448)

[3.2.2 Larger Scale Coordination 4](#_Toc322371449)

[3.2.3 Efficiency 5](#_Toc322371450)

# Introduction

In this document I will evaluate and discuss my views on: the overall project process, the products that were produced, the quality of the products and finally potential future expansion of the project.

## References

The Now Habit: A Strategic Program for Overcoming Procrastination and Enjoying Guilt-Free Play -- Neil Fiore

97 Things Every Programmer Should Know – compilation by Kevlin Henney

# Process

## Problems Encountered

I initially experienced a roadblock with creating the road network structure I needed for my simulation. I eventually overcame the problem, by breaking it down to creating a simple T network based on a four way cross intersection. From there, I was able to build out my network. I had spent a lot of time trying to find a tool that could be used for creating the network. I should have time boxed that research initially and then jumped into the manual approach.

I spent what seemed to me to be a lot of time with the formal specification. Late in the process I discovered that I had older version 2.5 (June 2009) of USE and the current version is 3.0.0 (Sep 2011). That could have made some difference. While working with the USE/OCL tool I feel like I lost focus on the problem/solution I was trying to specify because of focus on how to express my intent in OCL. Also, USE seems more geared towards the specification of object oriented problem-solutions and my project has a strong service oriented distributed asynchronous architecture. Perhaps some other formal tool would have been better suited? In the future I might consider a different tool for the formal specification. I would also time box that research and seek outside help/guidance from an expert.

I opted to try and use “cloud” resources for some aspects of the project. This caused problems when an internet connection was not available. I established a “Plan B” which was to make the notes I would normally make in the web based application in my local OneNote notebook. Later when connectivity was restored, I would updated the web application from my notes.

I was also a bit apprehensive about including some references in my public git repository because of copyright concerns.

## Estimates

Lorem ipsum

### Source Lines of Code

Lorem ipsum

graph

### Rework

Lorem ipsum

Graph

### Project Duration

Pie charts:

Overall Phase 1, 2 & 3 time

Overall time in activities

Time in activity for each phase

## Lessons Learned

Doing a formal specification did help me think about how the safety operations were going to work.

Doing upfront spikes and primers in the risk areas: SUMO, TraCI, Python, RabbitMQ and git was beneficial when it came to doing the implementation.

While working on this project I read the “Now Habit”. One of the big things I got out of this book is to focus on starting. Don’t get so overwhelmed with all that there is to do and become paralyzed by it. Ask when can I start working on the next task and what is the next task.

Having done a vision document with System Requirements was beneficial in that it contained the answer to, “what task is next?” Which also tied into the idea of “know your next commit” that I picked up from “97 things….”

Multiple git repos worked well

Future include OneNote nootbook in repo

Future secondary repo with tools

Being able to refer to a design helpful

While I had a Gantt chart, I really did more flow based task management.

Further into the project the cone of uncertainty narrows

Estimates refine/narrow as progress

Having a secondary machine was useful for spiking/testing alternate servers, etc.

Virtualization

Fast hard disk or SSD

# Products

## Quality

### Rework Ratio

Formula

actuals

### Test Coverage

Testing coverage slides

### Cyclomatic Complexity

#### Code Standard (PEP8)

### MTBF

## Future Work

Lorem ipsum

### More Generic

Lorem ipsum

### Configuration File

For agent RabbitMQ credentials

### Larger Scale Coordination

Lorem ipsum

#### Between Agents

Lorem ipsum

#### Scaled out regional networks/exchanges

Lorem ipsum

### Efficiency

Subscriptions to metrics

Reduce the number of exchanges and use topics