

Middle East Technical University

Department of Computer Engineering



CENG 492

Computer Engineering Design II

Configuration Management Plan

for TraffEdu

CodeSchbeke Software Solutions



Project Title: EDU3D

Advisor Asistant: Onur Soysal

Project Members: Sila Arslan

Çağla Okutan

Hatice Kevser Sönmez

Bahar Pamuk

Ebru Doğan

SPRING 2006

CONTENTS

1. INTRODUCTION

1.1 Purpose	3
1.2 Scope	3
1.3 Definitions and Abbreviations.....	3

2. SOFTWARE CONFIGURATION MANAGEMENT

2.1 Organization.....	4
2.2 SCM Responsibilities.....	5
2.3 Tools and Infrastructure	5

3. CONFIGURATION MANAGEMENT TASKS

3.1 Configuration Identification.....	6
3.1.1 Configuration Items.....	6
3.1.2 Baselines.....	7
3.2 Configuration Control.....	8
3.2.1 Requesting Changes.....	8
3.2.2 Evaluating Changes.....	8
3.2.3 Implementing Changes.....	8
3.3 Status Accounting.....	9
3.4 Configuration Audits and Reviews.....	10

4. PROJECT SCHEDULES AND CM MILESTONES.....10

5. PROJECT RESOURCES FOR THE CM.....11

6. PLAN OPTIMIZATION.....11

7. REFERENCES.....11

1. INTRODUCTION

1.1 Purpose

The purpose of this Software Configuration Management (SCM) Plan is to establish and maintain the integrity of TraffEdu system [1] software throughout its lifecycle. The SCM plan helps manage the configuration of the TraffEdu system and its components at specific points in time, control changes in the configuration, and maintain the traceability of the configuration during the entire lifecycle.

1.2 Scope

This Configuration Management Plan supplies the relevant policies and procedures for the development of TraffEdu. It describes how the configuration management elements of configuration identification, change control, status accounting, and audits and reviews shall be applied to establish the TraffEdu software configuration throughout the development effort.

This plan applies to all documentation, computer source code, executable programs, data files, software development tools, hardware, operating systems, and processes used in support of the implementation of TraffEdu system. Briefly, this document identifies the responsibilities and authorities for accomplishing the planned activities, it contains details on the items under configuration management process; it describes the required coordination of SCM activities with the other activities in the project and identifies tools and physical and human resources required for execution of the Plan. Finally it describes the planning of configuration management activities.

1.3 Definitions and Abbreviations

ABL: Allocated baseline

CCB: Configuration Control Board

CI: Configuration Items

CM: Configuration Management

CMP: Configuration Management Plan

CVS: Version Control System

C2S: CodeSchebeke Software Solutions

ERB: Engineering Review Board

FBL: Functional Baseline

MSN: Microsoft Network
PBL: Product Baseline
PL: Production Baseline
QM: Quality Management
SCM: Software Configuration Management

2. SOFTWARE CONFIGURATION MANAGEMENT

2.1 Organization

The organizational units related to SCM activities in the project are as follows:

- Configuration Control Board (CCB)
Reviews, denies or approves all the changes
Follows all the changes and ensures that they are done correctly
Approves all CI baselines
- Configuration Management (CM) Group
Creates and maintains the CMP
Coordinates and reports all the CM activities
Coordinates the development team for the changes
- Engineering Review Board (ERB)
Does the technical analysis and evaluation of the change plans
Evaluates the change proposals in the base of cost, time and performance and gives recommendation to the CCB
- Testing Team
Makes the change requests after the results of the testing process
- Developer Team
Creates the baselines and releases
Implements all the CM activities which are part of the actual system development

CCB is on top of the organizational units' hierarchy. All other units are supervised by CCB. The activities of the CM are followed by CCB and their request are either approved or denied. CM also coordinates the developer team. ERB evaluates the

change requests and gives recommendation to the CCB. Developer team implements all the change requests that come from CM.

2.2 SCM Responsibilities

In large projects, for CMP activities teams are divided into sub-teams which take different responsibilities, like program director, QM specialist, CM specialist, CM team, development contractor and testing team. But in such small projects like ours, all the members have to take responsibility in every part of the project. Our structure suggests that every member of the team is responsible for the part of the software that she will implement and will take all of the above CM organizational roles and responsibilities, during the life cycle of TraffEdu System.

The SCM activities and the corresponding organizational units are as follows:

SCM Activities	CCB	CM	ERB	Testing Team	Developer Team
Configuration Identification	Review	Originate			
Configuration Control	Review, Approve		Originate, Review	Test	Originate, Implement
Configuration Status Accounting	Review, Approve	Originate	Review		
Configuration Auditing	Originate, Review	Originate			

2.3 Tools & Infrastructure

As it can be inferred from the project schedule, modules of the system will be implemented concurrently. Although most of the time members will be working on different files, making changes on the same file will be needed. Also two of us will be working on the same module. In such situations there is need to merge the files and all the group members should be aware of those changes. In order to keep history of the source files we will use Concurrent Versions System (CVS) that is provided by department of Computer Engineering in Middle East Technical University. In addition, by using CVS the configurations to the source code and the documentation will be done in a sequence and the hazard of crashes will be avoided.

We are developing our project on Microsoft Visual Studio .NET environment. Before uploading a file to the CVS system, members will test their changes carefully on their own machines in order to prevent later conflicts. By applying this testing strategy there will be reasonable number of history files in CVS and managing them, will be much more efficient.

We do a meeting every week in order to discuss what difficulties we are encountering mostly and how can we overcome them. Also we discuss the progress of the team members. In addition, we use e-mail and MSN messenger communication in order to accomplish coherence between team members.

3. CONFIGURATION MANAGEMENT TASKS

3.1 Configuration Identification

Since software implementation is a continuous process, it passes through many steps as it gets closer to its final state. In order to keep track of its product's development, a project group must have a well defined procedure. In this section we will try to express our strategy to hold control of our product as it passes through inevitable changes.

3.1.1 Configuration Items

This section shows our agreed configuration items and processes for the items when they are configured.

Configuration items (CIs) identified for the project are:

- TraffEdu Configuration Management Plan
- TraffEdu Code Standards
- TraffEdu Requirements Data
- TraffEdu Design Data
- Source Code of the Project
- Executable Object Code
- Weekly Management Records
- TraffEdu Test Specifications
- TraffEdu Test Data

These components define behavior of the system and they will evolve through the development phase. “CVS” tool will be used for the configuration control and maintaining the changes of the project configuration items. Changes are automatically assigned to a revision number. The revision number along with the change is then stored in an associated archive.

A Configuration Item Index shall be created and placed under CVS server. It will be updated whenever a new CI or new version is established and posted to the shared e-mail of the development team (codeschebeke@gmail.com)

3.1.2 Baselines

Baselines are composed of CIs at a specific time in our progress. The baselines are used for controlling changes to the CIs throughout the life cycle of the project. C2S authorizes the creation of new baselines and approves any changes to a baseline. Baselines will be tracked, audited, retained and version controlled in the Concurrent Versions System (CVS) version control tool.

There are four different types of baselines for TraffEdu: functional, allocated, product and production.

The baselines and their contents are as follows:

Functional Baseline (FBL): A functional baseline is established following the determined requirements and delivery of Requirement Analysis Report. Every member of C2S is responsible of verifying all required documents and configuration items, related to project’s developments, are established properly.

Allocated Baseline (ABL): An allocated baseline is established at the end of the design phase. It is verified that all required documents and configuration items are established properly. The CIs in the allocated baseline include all preliminary system designs, detailed system designs, and associated test plans.

Product Baseline (PBL): A product baseline is established for each release of the system at the end of the test phase. The developers verify that the tested product is exactly described in the product baseline documents. It includes all CIs, CM data, system test plan reports, user manuals, and other plan documents.

Production Baseline (PL): A production baseline is operational when it is established at the end of each release or when a major change to the baseline takes place. It is established at the end of acceptance testing of that release. It includes all

CIs, CM data, system test plan reports, acceptance test plan reports, manuals, and other plan documents as the product baseline. The production baseline is the actual system release in use.

3.2 Configuration Control

Our configuration control activities are mainly composed of 3 parts which are significant for handling changes.

3.2.1 Requesting Changes

One of the members of CodeSchbeke Software Solutions may require a change in the previously implemented classes for enhancing the quality of the implementation or error correction. These changes are discussed with the other members via the shared e-mail of the team (codeschbeke@gmail.com) and in weekly meetings.

The change requests will be documented in the file *TraffEdu_Change_Request_File* and recorded by keeping the information of:

- The name(s) who made the request
- The version number where the problem occurs
- Date of request
- The explanation for the need for the change
- Description of the requested change

The change request records will also be available in CVS.

3.2.2 Evaluating Changes

All team members will discuss the change requests and consult to the advisor if necessary, by referencing our detailed design report and requirements in order to evaluate their effect on the whole project and see if it makes any conflicts with our baselines. At the end of the evaluation the change is approved or disapproved by the team and before the implementation of the change is started, the advisor will be informed with the change decision.

3.2.3 Implementing Changes

When the requested change is approved, the effects of the change on the other modules and the tasks of the members are revised. The version controlling will be

handled by CVS repository. The CVS repository stores a copy of all the source files and directories which are under version control. For the approved change, the implementation of effected modules will be committed to CVS and the version number will be changed accordingly. Since we are working with .NET environment, we will avoid committing the Release and Debug directories of the project to avoid wasting redundant space.

3.3 Status Accounting

CM status accounting is the recording activity of configuration management. This part of CM is responsible from providing current status of configuration items and documents. The status accounting records for TraffEdu are as follows:

- TraffEdu_Changes_Committed_File
- TraffEdu_Audit_Review_File
- TraffEdu Weekly Progress Reports

TraffEdu_Changes_Committed_File: This file is for the requested changes that are seen inevitable and so reflected to the design and implementation of TraffEdu. It will contain detailed description of the changes that are done. That is,

- The definition of the problem and how it was resolved
- Side effects of that change
- The person who requested the change
- The person who made the change
- Date of request of change
- Date of completion of the change
- Release and version

will be included in the file. The information will be held in an easy to handle text format and stored in CVS. The information will also be stored in the shared e-mail of project group where the discussion of change requests will be followed. The information first will be committed on the e-mail and then after approval of all members, will be transmitted to the CVS.

TraffEdu_Audit_Review_File: The result of the configuration audits and reviews will be stored in this file. The file will contain;

- The objective of the audit

- The CIs that are under audit and review
- The participants
- The date of the review
- Decisions

Like the *TraffEdu_Changes_Committed_File*, this file will also be stored both in the shared e-mail and also on CVS.

TraffEdu Weekly Progress Reports: These reports as the name implies are weekly, the reports will be both in the in the shared e-mail and also web-page of project team (<http://senior.ceng.metu.edu.tr/2006/codeschbeke>).

These status files will help us in releasing a new version; we will know what caused that new release of TraffEdu. They will also help us to find our position on the living schedule which will be provided on the web-page and will be updated weekly according to these status recordings.

3.4 Configuration Audits and Reviews

Configuration audits determine to what extent the actual CI reflects the required physical and functional characteristics [2]. So configuration audits are very important for the critical decisions of a project. For TraffEdu, main audits will be held before each release and the project meetings (minor audits) will continue to be held on every week as the first semester. The change requests, design and implementation problems, the things achieved that week, tasks of the next week and status of configuration items will be discussed in these weekly meetings. In main audits, the first thing to consider will be the achievement of design goals for that release. All the audits and reviews will be held by full participation of the project members. The result of these audits will be posted to the shared e-mail and also will be stored in *TraffEdu_Review_Audit_File*.

4 PROJECT SCHEDULES AND MILESTONES

Since the changes in a software project is inevitable, the project schedules will be updated weekly (at project meetings), according to the status accounting records of TraffEdu and latest schedule. The milestones of the project will be module completions (i.e. configuration items), integrations and documentation periods of system TraffEdu. The Gantt Chart will be prepared and put on the web site. We will

also have demonstration times after the integration of a completed module to the project members.

5 PROJECT RESOURCES FOR THE CM

The backup of the project will be achieved by using CVS system. The system will be used carefully in that some files due to their big content and that doesn't need much changes (like the resources files) will not be saved on CVS at each version of TraffEdu, rather will be stored in the shared e-mail. All the changes to the files in CVS system will be made according to the decisions of the project members. Such decisions will be handled in the meetings or through the shared e-mail.

6 PLAN OPTIMIZATION

During the development progress this CMP may subject to change. Since the team has no unique continuous leader for the project, all the project members are responsible from monitoring the overall development cycle, that is CVS activities. If the project is behind the schedule at some time, new processes are needed or a design trouble comes up, a plan optimization will be inevitable for TraffEdu.

7 REFERENCES

- [1] C2S Requirement Analysis Report Section 1.4
- [2] IEEE Standard for Software Configuration Management Plans (IEEE Std 828-1998)