

MIDDLE EAST TECHNICAL UNIVERSITY
DEPARTMENT of COMPUTER ENGINEERING

CENG 492-COMPUTER ENGINEERING DESIGN 2

'BluePost'

CONFIGURATION MANAGEMENT PLAN

by



Duygu CEYLAN – e1394782
Ertay KAYA – e1356948

Seda ÇAKIROĞLU - e1394816
Hüseyin ÖĞÜNÇLÜ - e1395318

Gözde ÖZBAL – e1395326

TABLE OF CONTENTS

TABLE OF CONTENTS	2
1. INTRODUCTION	3
1.1 Purpose of the Configuration Management Plan	3
1.2 Scope of Document	3
1.3 Definitions and Abbreviations	3
1.4 Document References	4
1.5 Document Overview	5
2. THE ORGANIZATIONS CM FRAMEWORK	5
2.1 Organization:	5
2.2 Responsibilities	6
2.3 Tools and Infrastructure	7
3. THE CM PROCESS	7
3.1 Identification	7
3.2 Configuration Management and Control	8
3.3. Configuration Status Accounting	10
3.4. Auditing	11
4. PROJECT SCHEDULES – CM MILESTONES	11
4.1 Project Schedule	11
4.2 CM Milestones	12
5. PROJECT RESOURCES	12
6. PLAN OPTIMIZATION	13

1. INTRODUCTION

1.1 Purpose of the Configuration Management Plan

During the development process of a software product, a group of people take roles and work concurrently. For a team to accomplish the tasks involved in the development process in a more effective manner, planning is necessary. Throughout the development phase, some changes and updates may be necessary at certain stages. Then, these changes must be fully documented and all of the team members must be informed of these changes to avoid any possible inconsistency. Good planning enables an effective communication between the members and eases the process of controlling the development process. Thus, configuration management plans are prepared to define the process of identifying, managing, and auditing the changes as they occur.

In a similar manner, this plan serves the purpose of managing the configuration of the “BluePost” Project throughout its lifecycle, which is conducted by a team of five members.

1.2 Scope of Document

This document is prepared to handle the configuration management activities of the “BluePost” Project. The activities discussed here are applicable to all documentation, source code development, software and hardware tools used, and any other process involved.

This document clearly explains the responsibilities the group members have. The process of identifying the necessary changes and updates, management of these changes, and informing the other team members of the changes is described. During the development of the project, we will follow steps in order to avoid any misconflict about the current status of the project. In other words, this report is intended for the group members in first place. Our supervisor and the instructor are among the other audience of the document.

1.3 Definitions and Abbreviations

- CCB: Configuration Control Board
- CI: Configuration Item
- CM: Configuration Management
- CMP: Configuration Management Plan

- CR: Change Request
- CVS: Concurrent Versioning System
- CSA: Configuration Status Accounting
- SCM: Software Configuration Management
- SCMP: Software Configuration Management Plan
- SCR: System Change Request

Configuration Control Board represents the group of people who accept or reject the proposed changes.

Configuration Items are the things to which configuration management can be applied such as the documents describing the development of the product, or modules.

Auditing describes the process of determining whether the developed parts of the product satisfy all the requirements or not.

Configuration Control represents the process of request, evaluation, approval, and implementation of changes to configuration items.

Baselines are used as a reference during the lifecycle of the project such as the design reports, snapshots, and demos.

1.4 Document References

While preparing this plan, we used the following documents as a reference:

- IEEE Standard for Software Configuration Management Plans (IEEE Std 828 - 1998)
- Software Configuration Management Plan , Presentation prepared by METU Computer Engineering Department for the course Ceng 492
- 2004 – 2005 METU CENG Senior Project Groups Documents,
<http://senior.ceng.metu.edu.tr/2005>
- 2005 – 2006 METU CENG Senior Project Groups Documents,
<http://senior.ceng.metu.edu.tr/2006>

1.5 Document Overview

This document consists of six main parts, each of which is described briefly below:

- **Introduction:** The introduction part explains the purpose of preparing a CMP and the scope of this plan. In addition, the abbreviations and terms are defined as well as the references which have been used.
- **The Organizations CM Framework:** This part explains the responsibilities of the group members for accomplishing the CMP. The tools that are going to be used are included as well.
- **The CM Process:** The identification, management, and the auditing of the CIs are discussed here.
- **Project Schedule – CM Milestones:** The deadlines for the CM activities is given in this part.
- **Project Resources:** The resources that will be used for CM activities are described in this part.
- **Plan Optimization:** This section explains the methods that can be used to optimize the CMP.

2. THE ORGANIZATIONS CM FRAMEWORK

2.1 Organization:

As AlinYazilim team members, we have formed several organizational units related to the SCM activities in the BluePost Project.

- **Configuration Control Board (CCB)**

All the CM activities are handled by this board. All other units are supervised by CCB.

The responsibilities are:

- i) Reviewing, denying or approving the changes
- ii) Following the changes and ensuring that they are done correctly
- iii) Approving all CI baselines
- iv) Holding audits
- v) Ensure implementation of SCRs
- vi) Update CM schedule

- **Configuration Management Group**

This group is responsible for creating and maintaining the CMP, coordinating and reporting all the CM activities and also coordinating the development team for the changes.

- **Engineering Review Board (ERB)**

This board does the technical analysis and evaluates the change plans and also evaluates the change proposals in the base of the cost, time and performance, gives recommendation to the CCB.

- **Testing Team (TT)**

This team is responsible for making the change requests after the results of the testing process.

- **Developer Team (DT)**

This group of people creates the baselines and releases, implements all the CM activities that are part of the actual system development.

- **Version Control Team (VCT)**

VCT is responsible for following the version of the product.

- **Release Control Team (RCT)**

RCT is responsible for controlling the releases of the project.

2.2 Responsibilities

Because of the fact that BluePost is developed by only five members, all the people in AlinYazilim are members of the CCB. The responsibilities of each member of the AlinYazilim team for CM are as follows:

1. Performing the assigned CM activities
2. When some change is done in CVS, informing all the members
3. Completing the assigned jobs until deadline, that is conforming to the CM schedule
4. Informing the people about an SCR by sending the proper e-mail
5. Depending on the CM process, checking in and checking out by using CVS properly
6. Commenting on the changes properly, while committing to CVS

2.3 Tools and Infrastructure

The modules of the BluePost Project have been implemented concurrently so far and most often, several members working on the same module or the same file occur. As a result, each member should be aware of the changes made by the others. In order to keep track of the history of the source files, Concurrent Versioning System (CVS) will be used. This system allows text and binary files to be versioned in a central repository and allows concurrent development of the files in each module. We are assured that being capable of seeing any previous version of a file will be really useful during the development of our project and the hazard of the crashes will be avoided. CVS will be provided by the department of Computer Engineering in Middle East Technical University. As for the CVS client, we will use Eclipse, which is a fully integrated development environment for Java, because of the fact that Eclipse is a very preferable CVS tool which works under both Linux and Windows.

3. THE CM PROCESS

3.1 Identification

The ‘BluePost’ project can be identified as a combination of the following configuration items (CIs): Software, Hardware, Data Transfer, Documents, Baselines

1. Software CI:

This CI is related to “File Upload” and “Format Conversion” modules. Both of these modules were designed according to Object Oriented Design methodologies and written in Java. When any part of this CI is changed, corresponding UML diagrams in the design are also changed accordingly. These changes are also done obeying the syntax specifications determined before.

2. Hardware CI:

This CI is related to “Register Process” and “VGA Process” modules. The circuits that implement these modules are described using VHDL. These circuits are documented using schematic representations according to the entities they include and when a configuration is done in the VHDL code the corresponding block diagrams are also

changed in the design. These changes are also done obeying the syntax specifications determined before.

3. Data Transfer CI:

This CI is related to the remaining modules and consists of the following sub-items

- a. **Serial-Parallel Communication:** A printed circuit board (PCB) and a C program loaded on the board are used in this item. The details of the board design and the program are documented and this documentation will also change in case of an alteration in the board design or in the C code.
- b. **Wireless Bluetooth Communication:** This sub-item is related to the AT commands used for Bluetooth Evaluation Kit. In case of any change in the structure of these commands, the documentation that specifies these structures will also change accordingly.

4. Documentation:

‘BluePost’ project has a detailed documentation that can be referred efficiently in case of a change in the modules and can be easily updated after a module configuration. The following reports are prepared so far:

- Project Proposal
- Requirement Analysis Report
- Initial Design Report
- Detailed Design Report
- Configuration Management Plan (CMP)

5. Baselines:

Baseline CIs are related to the milestones of ‘BluePost’ and can be divided as followings: Requirement Analysis, Project Design, Project Prototype, CMP, Implementation Completion, Testing, Documentation

3.2 Configuration Management and Control

During the development of the ‘BluePost’ project, source configuration is done by following some steps that make easier the control of the corresponding configuration process. We

decided to follow different steps for each CI since the change process will vary from item to item.

The steps to be followed in the configuration of CIs are as follows:

1. Change Request(CR):

When a team member feels the requirement of changing any module of the project s/he sends an e-mail to the mail account of the team with a subject line “CHANGE REQUEST: *Module Name*”. A .doc file is attached to the mail that has a strict format explaining the details of the request. The file format is as follows:

- ID of the CR
- Date of the CR
- Name of the team member that makes the CR
- CI and sub-item (if exists) of the CR
- Module related to the CR
- Urgency of the CR ranging from 1 to 5
- Description of the CR:

If the CR is related to the Software CI, the file(s), class(es) and the function(s) are specified in this part.

If the CR is related to the Hardware CI the file(s), entity(ies) and the architecture(s) are specified in this part.

If the CR is related to the serial-parallel communication sub-item of Data Transfer CI and about the C code the file(s) and the function(s) are specified in this part. If the CR is about the PCB of this sub-item the parts of the board to be changed are specified in this part.

If the CR is related to the Wireless Bluetooth Communication sub-item of Data Transfer CI, the files and the AT Commands to be changed are specified in this part.

Finally, detailed explanation is made why the change is necessary.

2. Change Request Evaluation:

The group members are informed with the CR by checking the mail account regularly and they make an investigation about the request and determine how the change will affect the rest of the sources and how will it be done. After these individual

investigations all group members discuss the request in weekly meetings. If an approval results after discussion one or more group members take the responsibility according to the knowledge they have on the corresponding module to complete the change by the date determined in the discussion. If the CR is disapproved the team member that made the request adds the result to the .doc file in the mail account and stores the file in the directory used for configuration control naming it as *CREvalDate_ModuleName.doc*. If there are more than one CR to be discussed in a meeting, the discussions are made ordered in the urgency levels of the CRs.

3. Change Implementation:

After the approval of a CR, assigned group members make the agreed changes by the specified date. Afterwards they test and make sure that the changes work properly with the whole system. The changed files are put into the CVS repository supplied by our department with their new version names. Same members also reflect the changes in the documentation. If the change is to be done on the printed circuit board, all group members are involved in the change process and they consult to an expert before doing the change. Finally, the team member that made the request adds description to the .doc file in the mail account about the change process and stores the file in the directory used for configuration control naming it as *CREvalDate_ModuleName.doc*.

3.3. Configuration Status Accounting

Configuration Status Accounting (CSA) aims to inform the group members about the modifications and updates of the project modules. By CSA, status of configurations on CIs in the operational inventory can be followed by members easily and a synchronized pattern can be formed. Near the group members, by this way we find the chance of declaring the status of our project to our teacher and assistant.

For modifying any module of our project, the member that makes the modification should;

- Put the modified source code in related part of CVS,
- Include very detailed report about why modification is needed,
- Provide the group members with a well structured documentation about how the module is modified,
- Check all the project again for preventing conflicts and system errors,

- Take the control of the modified part and deal with the problems if any one occurs,
- Arrange every modification according to their dates by numbering them,
- Announce the new status.

For announcing configuration status, we will use our website, emails, weekly reports, weekly meeting and CVS.

3.4. Auditing

Software Configuration Audit is one of essential parts of our project. It basically contains several independent checking processes for the newly integrated functionalities and modified project components. It aims to preserve completeness and consistency of project modules.

After every modification, auditing stage takes place regularly. The group members continue auditing until the configuration status is capable as we declared in our reports. We will control the changes for being sure about their effectiveness and accuracy for our project functionalities. By audits, the group members understand the differences between baseline configuration and current configuration and make decisions about the necessities of the status of configuration.

We decided to use two basic steps for auditing;

- Functional Configuration Audits(FCA):
It aims to understand if every SCI is well-formed and the version has all the necessary functionalities.
- Peer Reviews(PR):
It aims to share the thoughts of the other people that use the project and understanding the needed modifications that seem as necessary by the users.

4. PROJECT SCHEDULES – CM MILESTONES

4.1 Project Schedule

The Living Schedule we have prepared in the beginning of the second term clearly identifies all the tasks that need to be done and the important deadlines that need to be caught through out the term. All the group members will follow this schedule and work on the tasks assigned to them for a certain week. In addition to the weekly meetings with the supervisor, we will

conduct meetings to discuss the current status of the project. By the means of these meetings, we will inform each other about our progress so that we can make sure the schedule is followed accordingly. The Living Schedule which is located in the project website will be updated on a regular basis.

4.2 CM Milestones

The milestones for our project, as documented in the Living Schedule and described in the 3.1 *Identification* section, are listed below:

- CM Plan Delivery 11.03.2007
- Audit of Software CI 15.03.2007
- Audit of Hardware CI 08.04.2007
- Audit of Data Transfer CI 29.04.2007

As we progress to catch up with these milestones, the necessary documentation will be prepared as well.

5. PROJECT RESOURCES

In order to apply our CMP and follow the CM activities, we will use CVS. Since our team consists of five people, it is important to work on the same source without inconsistencies. By the help of CVS, the changes and updates to the source are handled easily and new versions are produced as a result of these situations without losing the older versions.

In addition, in our web page we will keep all project documentations and living schedule which is being updated continuously which will help us to follow the CM activities easily.

Moreover, we are using Xilinx Web Pack for to compile our VHDL code and we are using Hyper Terminal tool of Windows XP to communicate with our Bluetooth board before having a parallel to serial converter.

Finally, since all team members will be responsible for keeping track of the changes and informing all other members, ALINYAZILIM Team constructs the most important part, human resources part of the project.

6. PLAN OPTIMIZATION

The configuration management plan will be a guide for our team in the development of “BluePost”. However, there can be some changes during the development process and in such cases we will make necessary optimization in our configuration management plan and in our progress. Communication between the team members is an important issue in order to be aware of the changes and follow the plan. We are a small group of 5 people and we have weekly meetings with group members and with our supervisor. As a result, it will be easier to follow the changes and make necessary optimizations. All team members will be responsible for being aware of the updates and changes in the system and informing the other team members. For this purpose, team members may use e-mails for communication. By this way, we will keep track of changes and delays in our progress schedule and make our plan optimization frequently.