MIDDLE EAST TECHNICAL UNIVERSITY

COMPUTER ENGINEERING DEPARTMENT

CEYLAN©
AJAX SOFTWARE DEVELOPMENT STUDIO

CONFIGURATION MANAGEMENT PLAN

Group Members:

Saliha ALTUNSOY

Candan CEYLAN

Canan ESKİ

Yavuz GÖKIRMAK

Duygu GÖRGÜN

Table of Contents

1. Introduction....3
1.1 Purpose....3
1.2 Scope and Overview....3
1.3 Definitions, Acronym and Abbreviations....3
1.4 References....4
2. The Organizations CM Framework....4
2.1 Organization....4
2.2 Responsibilities....5
2.3 Tools & Infrastructure....6
3. The CM Process....7
3.1 Identification....7
3.1.1 Modules....7
3.1.2 Code....8
3.1.3 DataFiles....8
3.1.4 Documentation....8
3.2 Management and Control....9
3.3 Configuration Status Accounting....10
3.4 Auditing....11
3.4.1 Functional Audits....11
3.4.2 Physical Audits....11
3.4.3 Process Audits....12
4. Project Schedules – CM Milestones....12
5. Project Resources....12
5.1 Personnel....12
5.2 Tools....13
6. Plan Optimization....13

1. Introduction

1.1 Purpose

The purpose of this Software Configuration Management (SCM) Plan is to provide the continuity of our project Ceylan, when we encounter the changes and updates during development of Ceylan. In a project like ours, making modifications can not be easy and sometimes these modifications can confuse the project members. This SCM Plan helps us to identify, organize and handle these changes/modifications in development of Ceylan appropriately in order to prevent this confusion.

1.2 Scope and Overview

The main scope of this document is to define SCM Plan and apply it to our project Ceylan. In this document we will first mention the organization and responsibilities for the configuration management and the tools and infrastructure used for configuration management. Then we will describe the configuration management process in mainly four topics namely, identification, management and control, configuration status accounting and auditing. And finally, we will talk about project schedule & configuration management milestones, project resources and plan optimization.

1.3 Definitions, Acronym and Abbreviations

API	Application Programming Interface
CCB	Configuration Control Board
CI	Configuration Item
CM	Configuration Management
CVS	Concurrent Versioning System
FTP	File Transfer Protocol
GUI	Graphical User Interface

SCM	Software Configuration Management
SCR	Software Change Request
XML	Extensible Markup Language

1.4 References

This Configuration Management Plan is prepared by the help of the following formats describe in:

- IEEE Std 828-1998 IEEE Standard for Software Configuration Management Plans (Revision of IEEE Std 828-1990)
- Configuration Management presentation prepared by Middle East Technical (METU) Computer Engineering Department for the course CENG492.
- “Software Engineering: A Practitioner's Approach” by Pressman, Roger S. (2001): Fifth Edition.
- http://en.wikipedia.org/wiki/Software_configuration_management

2. The Organizations CM Framework

According to the needs of our software development project, the organization of team and the relevant roles are defined. Our team consists of five students and because of this; the roles may have more than responsible student. In addition, if there occurs any problem, the roles may change. More detailed information is given at 2.1 Organization.

2.1 Organization

First of all, all of the team members are also a member of Configuration Control Board (CCB). The main responsibilities of CCB are:

- Establish and control baselines
- Check CVS
- Perform the assigned CM activities
- Inform all members when a change is done in CVS

- Attach comment after changed codes in the CVS
- Finish the jobs until deadlines
- Form CM policies
- Identifying the configuration items
- Deal with the time and resource planning issues
- Check the CVS usage activity of members
- Check the correct actions to the Software Change Request (SCR) are done or not

In order to avoid problems related to the responsibilities that the team member have to perform when he/she is assigned to handle a task, roles must be identified clearly. The SCM activities also have to be identified clearly.

Moreover, we have to assign more detailed responsibilities to the members in order to get the most efficient result. So, some groups are built up:

- Quality control
- CM Update
- Team control
- Testing
- Development

More detailed information about these groups will be given at part 2.2.

2.2 Responsibilities

All members have to follow their responsibilities. The organization and roles of the team are as the following:

- ✓ Quality control team:
  - Be careful about the standards
  - Ensure that the software development process steps are done according to CMP
  - Ensure that the correct product is released
  - Ensure that members are doing their work properly

- ✓ CM Update team:

- Update the CM schedule according to the works done
- Report the CM activities whether they have done properly or not

✓ Team control team:
- Ensure that the change control are performed
- Maintain CMP
- Control releases

✓ Testing team:
- Test the releases of the project
- Report errors
- Tests the results of the changes done related to SCRs

✓ Development team:
- Create the baselines and releases
- Perform developmental change control activities
- Use CVS for update of code to the corresponding folder.
- Implement the product
- After correction of errors, implement the product again

2.3 Tools & Infrastructure

We are going to use CVS as our version control system. CVS server is supplied by our department. CVS keeps a central repository that includes current source code, past versions of the system, and logs that document changes to the system. This is good for source codes; being able to follow the changes that are made to source code by other developers is very critical for the success of the project. Also, any previous version of a file can be seen by the help of CVS. Besides these, it can be controlled that whether the member made any change on a file that has to be updated and which parts of the file was changed. We could see it due to the property of CVS that merges the old file and new file, and also put marks on the parts on which there is any change.

In order to use CVS, team members use two different ways. One of them is Eclipse. There is a user interface for connecting to CVS and do work about files. There are some basic reasons why we choose Eclipse: We use Eclipse during the project and Eclipse run

in both Windows and Linux. On the other hand, the other way is SSH protocol. This is also preferable way for team members that are used to using shell.

3. The CM Process

The Configuration Management process consists of functions and definitions of the tasks required to manage the configuration of the software. In order to administrate the CM, both technical and managerial SCM activities must be identified. These SCM activities are generally divided into four parts:

- Identification
- Management and Control
- Configuration Status Accounting
- Auditing

These categories are explained in details below.

3.1 Identification

In order to identify the current state of our project Ceylan, we defined some Configuration Items (CI). We have chosen these items as configuration items because they are the most important parts that are prone to changes and Ceylan's status is going to be determined according to these items. Our CIs are listed below:

- Modules
- Code
- Data Files
- Documentation

3.1.1 Modules

Modules of our project are defined in the design stage. These modules are determined by grouping the similar functionalities together. However, need for some changes in these modules may arise in the implementation stage. For instance, we may see that one more module should be added in order to add functionality to Ceylan or we may see that one package should belong to another module.

3.1.2 Code

Code is the most changeable configuration item in our project. From the beginning to the end, there will be changes in the code continuously of course. However, once the standards are defined the new codes will be generated accordingly. Our coding standards can be listed as below:

- The variable names should be meaningful.
- The function names should be meaningful and should reflect the responsibility of that function.
- If the names of the variables or functions are combinations of more than two words, the first letters of all the words but the first one should be capitalized.
- Comments about the arguments and the responsibility of the functions should be added before the function definitions.
- Comments about the responsibility of classes should be added to the top of the class definitions.
- The class names should be meaningful.

3.1.3 DataFiles

Data files are the XML files created to store necessary information for the flow of Ceylan such as the path information about workspaces and projects, etc. The structure of these XML files may change because of the additional properties that are needed to be stored.

3.1.4 Documentation

Detailed information about Ceylan should be documented as a necessity for the software management and as an assistant for us-the developers. Up to now, we have written the following documents:

- Project Proposal
- Requirement Analysis Report
- Initial Design Report
- Final Design Report

These documents are kept in our group's storage and will be used as an assistant when needed. In addition to these reports, we are going to generate a Test Specifications report and document the release notes.

3.2 Management and Control

We have built a Wiki in order to control the progress of our group. In this area, we had created tables to list information about the jobs to be done, the responsible member from the group and the deadlines. Since this area is accessible and modifiable by all group members, the announcements are made through filling in tables here and information change requests are gathered again through forms from this page. All members are responsible from evaluating the request and determining the affects of the change for their modules. These change requests, their priorities and expected affects are discussed in our weekly meetings. If the request is decided to be taken into account, the result is announced through Wiki and responsibilities are assigned to members in order to make the changes effective. Responsible members make the necessary changes to code and upload it to CVS so that all members can reach it. The project manager is responsible from controlling the completeness of the change process.

The Software Change Request form which is in our Wiki can be seen below:

Software Change Requests

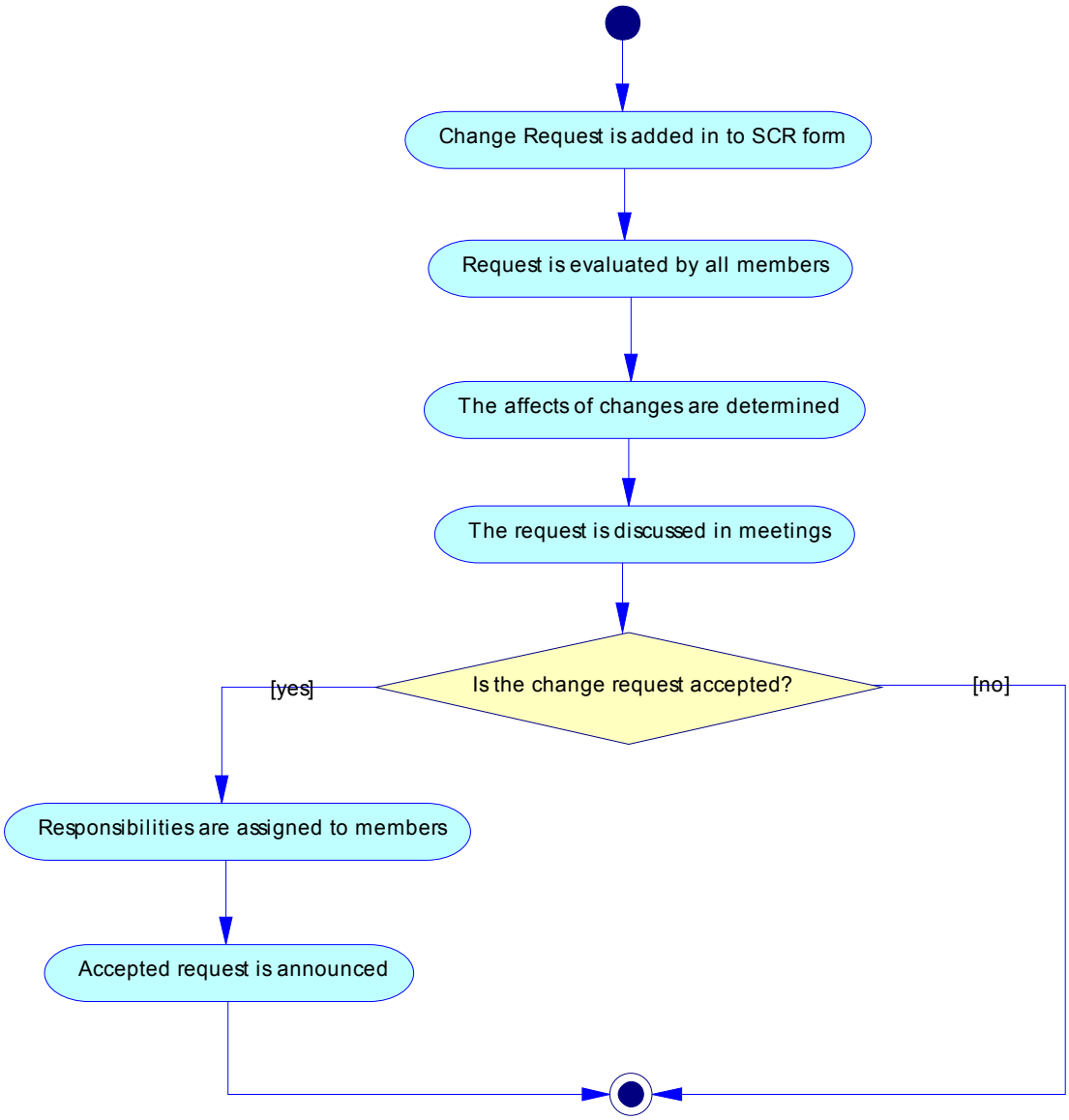
Request ID	Request Date	Change Deadline	Request Owner	Description	Related Module
01	05.03.2007	23.03.2007	Saliha	Modularization changes	All Modules

The announcement board for accepted requests can be seen below:

Accepted Change Requests

Request ID	Acceptance Date	Deadline	Assigned Members	Related Modules
01	09.03.2007	25.03.2007	All Members	All Modules

The process describing the change requests and evaluation steps is described through the activity diagram which can be found below:



3.3 Configuration Status Accounting

Configuration Status Accounting mainly is about recording and reporting the status of the configuration items and changes applied to them. The status of the project will be seen in the Living Schedule added to our web site. Moreover, reports about status will be constructed regularly including information about changes and revisions. The name of the files subjected to changes, the reason for changes, version numbers and the name of the

group member who made the change will be included in reports. These reports will be accessible from the Google-groups's web page that we had created for file sharing. Additionally, one copy of them will be stored in the CVS repository if possible.

3.4 Auditing

In order to develop software that has high quality and meet the requirements specified in the analysis phase, Software Configuration Auditing should be done. The auditing which is done regularly will give us the chance to control the impact of changes so that necessary precautions can be taken. We are going to discuss these again in our weekly meetings and we will arrange special meetings after important changes and when necessary.

We are going to follow an auditing process consisting of Functional Audits, Physical Audits and Process Audits.

3.4.1 Functional Audits

The objective of a functional audit is to provide an independent evaluation of software products, verifying that its configuration items' actual functionality and performance is consistent with the requirement specifications. We are going to hold this type of audit prior to the software delivery to verify that all requirements specified in the Software Requirements Specification have been met. Before release of the final package, we have to make sure that we have implemented all the requirements. Some functional requirements that are specific to our project and to be verified are:

- The proper functioning of database, CVS and FTP features
- The proper functioning of editor part which consists of syntax highlighting, synchronization of code and design views and drag & drop features etc.
- The proper functioning of menu bars, tool bars, widgets etc.
- The proper functioning of debugging process

3.4.2 Physical Audits

This type of audit is a formal examination of the existing configuration items to verify that requirements, specifications, and standards applied in producing the configuration

item have been addressed and all features delivered can be traced through the development process and through technical requirements. We are going to make physical audits after the release of final package.

3.4.3 Process Audits

The process audit is concerned with the validity and overall reliability of the development process itself. In this type, the important question is: Is the process consistently producing acceptable results? Since the development process is continuous, we are going to make this kind of audit regularly during the implementation stage.

4. Project Schedules – CM Milestones

We have a Living Schedule on our web site and neccessary changes will be made on it during the development.

Team members have meeting with supervisor on Thursdays every week. In these meetings members and supervisor discuss about current level of the project. After this meeting Anafor members make a meeting for coming week, each member gives information about her/his part.

Ceylan IDE composed several modules as database, FTP, drag&drop etc.. When a module is finished it will be integrated to the system. We will follow CM activities while integrating the modules. Versional changes will be made according to CM activities. In addition to module integration, we will perform CM activities while testing phase.

5. Project Resources

5.1 Personnel

Human resources of the configuration management are Anafor Team members.

Saliha Altunsoy is the team leader and manages the overall configuration process of the project. Together with Yavuz Gökırmak, she is responsible from the changes made in editor and error handler. Yavuz Gökırmak also manages the changes made in FTP and Database modules. Candan Ceylan, Duygu Görgün and Canan Eski are responsible from changes made in GUI. They will also manage documentation and test scenario changes.
All team members will provide API’s for the modules that they are reponsible from. As a result when there is a change in a module, the other ones will not be injured.

Team members have educational and applicational background of configuration management from the course CENG350 that they have taken last year.

5.2 Tools

The team will use CVS as the main Configuration Management tool. It is provided by our department (METU – Computer Engineering Department).

Without using CVS, there appeared important change inconsistencies in our work in previous semester. Because, it is hard to maintain the code with the changes added by each 5 team members. Therefore, CVS is very important in means of reliable configuration management. All team members will be able to keep up with the changes made on the code.

The teams' mail group is also another important tool for configuration management for two main reasons. Firstly, team members are living apart from each other and have different schedules. So it is not possible to make a meeting each time there is a change. Mail group serves as an online meeting platform. Secondly, the reasons and details of the changes are explained and discussed in the mail group.

6. Plan Optimization

All the project members are responsible for monitoring and maintaining the CM Plan. In our weekly meetings, we can evaluate the changes can be done to the Plan and if it is essential, we can approve these changes and update our CM Plan weekly. Moreover, these changes can be made using CVS and project members can be informed via e-mail using mail group.