Software Change Management

Unit 4

Red: indicates important

Contents

- Baselines
- Software change management
- Version control
- Change control
- Auditing and reporting

Baselines

- Configuration items are independently tested, stored , reviewed and changed.
- Software configuration item can be one or more of the following :
  - System specification
  - Source code
  - Object code
  - Test plan and test cases
  - Design data
  - Project plan
  - Standard procedures
  - Database schema and file structure
  - Product specific document
- Baseline : It's a set of configuration items that have been formally reviewed and agreed upon that there after serves as the basis for future development, and can be changed through formal change control procedures.

- Its an approved software configuration item that has been reviewed and finalized.
- Example for baseline: approved design document that is consistent with the requirements.
- Its functionally complete.
- It can be recreated at any point of time.

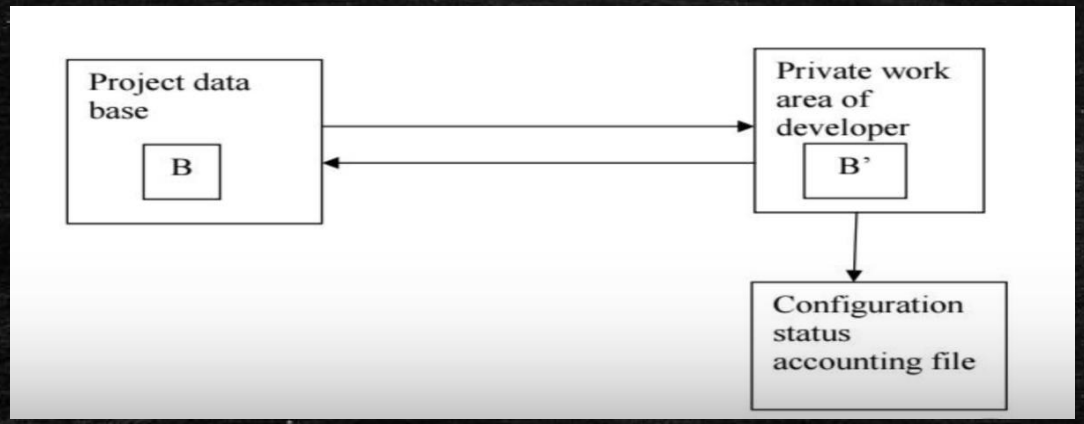
Software Change Management

- Its an activity that is applied throughout the SDLC.
- Its the task of tracking and controlling changes in the software.
- Software Configuration Management(SCM) is the discipline for systematically controlling the changes in software and supporting documents (like Test Cases, Test Plan, Design Documents, SRS etc.) during the software development life cycle.
- Its aim is to maintain integrity of the software while including changes.
- Objectives of Software change management process:
  - Configuration identification: Identifying configurations, configuration items and baselines. It involves identifying each component name, giving a unique version name.
  - Configuration control: Controlling changes to a software. It's a process that controls the release of the product and the changes.
  - Review: Reviewing the process to ensure consistency among different configuration items. Its involved in evaluating consistency and completeness of the software.

- Status accounting: Recording and reporting the changes and status of the components.
- Auditing and Reporting: Validating the product and maintaining consistency of the product throughout the SDLC.

Process of changes

- Baseline forms the reference for any change.
- Whenever a change is identified, the baseline in the project database is copied by the software developer to his private area. Once the modification is being done the baseline is locked for any further modification which makes it inconsistent. The records of all changes are recorded through a change control procedure, it becomes a approved item for updating the original baseline in the project database.

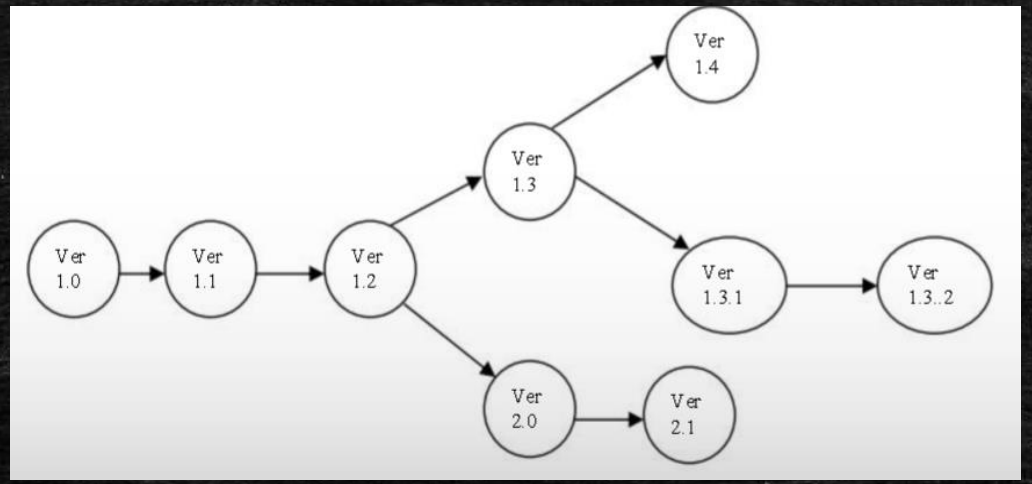


- All the change during the process of modification are recorded in the configuration status auditing file. It records all changes made to the previous baseline B to reach the new baseline B′. Status accounting file ensures that the new baseline B′ has all the required approved changes incorporated. This is also known as auditing.

Version Control

- It's the management of multiple revisions of the software system during development process.
- Aka revision control software, source control software.
- For example, a SRS is produced including the user requirements which change with time. Once the SRS is finalized, documented and approved, its given a document number, with a unique identification number. The name of the items follow a hierarchical pattern which consist of the following:
  - Project identifier
  - Configuration item
  - Change number or version number

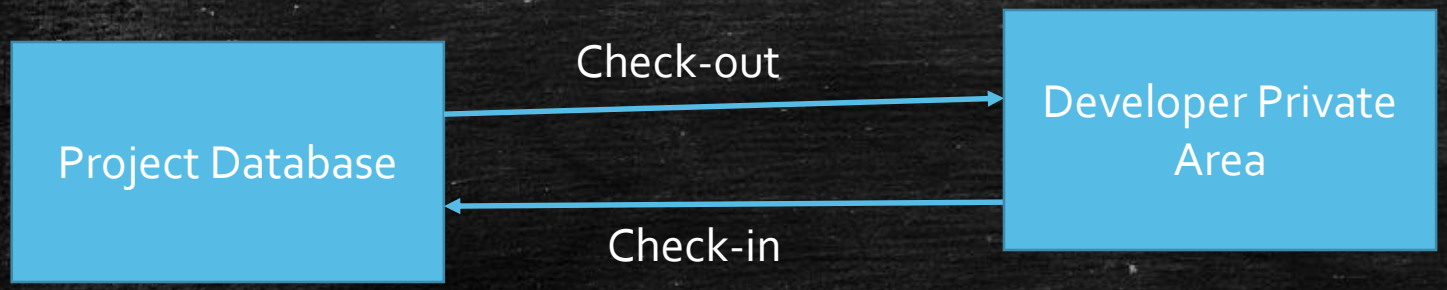
- The identification of configuration item must provide relationship between items whenever such relationship exists. The identification process should be such that it uniquely identifies the configuration item throughout the development life cycle. An evolutionary graph reflects the history of all changes.
- The aim of these controls is to facilitate the return t any previous state of configuration item in case of any unresolved issue in the current unapproved version.
- E.g. of evolutionary graph :



- The above figure is an evolutionary graph that represents evolutionary graph of a configuration item during the development life cycle. Initial version of the item is given version number Ver 1.0. Subsequent changes to the item which can be bug fixing or adding minor functionality is given number Ver 1.1 and Ver 1.2. After that, a major modification to Ver 1.2 is given a number Ver 2.0 at the same time, a parallel version of the same item without major changes is maintained and given a version number Ver 1.3.

- Version numbers are given by version control manager to identify the configuration item throughout the development lifecycle. It depends on the volume and extent of changes. Versions of the items are released during the maintenance phase.
- Software engineers use version control mechanism to track the source code, documentation and other configuration item. Many tools are there to store and number these configuration items automatically. As software is developed and deployed, it is common to expect that multiple versions of the same software are deployed or maintained for various reasons.
- It is also sometimes desirable to develop two parallel versions of the same product where one version is used to fix a bug and the other is used to add new functionality and features.
- The project database maintains all copies of the different versions of the software and other items. Updating to the central project database after completing changes will lead to overwriting of each others work. Most version control systems provide a solution by locking the version for further modification.

- Commercial tools are available for version control which performs one or more of following tasks:
  - Source code control
  - Revision control
  - Concurrent version control
- There are many commercial tools like Rational ClearCase, Microsoft Visual SourceSafe.
- Once the code is tested and finalized, the first step is to register the program to the project database. The revision is numbered and this file is marked read-only to prevent any further undesirable changes. This forms the building block of source control. Each time the file is modified, a new version is created and a new revision number is given.
- The first version of the file is numbered as version 1.0. Any further modification is possible only in the developer's private area by copying the file from the project database. This is called as check-out.



Changes to a baselined file in project database

- Version control process starts with registering the initial versions of the file. This essentially enforces a check on the changes which ensure that the file can't be changed unless it is checked-out from the project database. When a change is to be made, the developer must extract the latest version of the file from the project database. Once its checked-out from the project database for modification, the file locked. Other developers can check-out the file for read-only purpose.

Change Control

- Change control is a systematic approach to managing all changes made to a product or system. The purpose is to ensure that no unnecessary changes are made, that all changes are documented, that services are not unnecessarily disrupted and that resources are used efficiently.
- Changes can be initiated by the user or other stake holders in the maintenance phase. A change request may even arise during the development phase.
- A change request is beginning of any change control process. The merits and demerit of change request is evaluated, and the side effects are evaluated. A change control report is generated by the technical team. It lists the extent of changes and potential side effects. A team called change control authority makes the final decision, whether to accept or reject the change based on the change control report.
- Engineering change order is a change order that is generated after the approval of the change request by the change control authority. It forms the starting point of effecting a change in the component.

- If the change requested is not approved, then the decision is conveyed to the user or the change request generator.
- When change order is received by the developers, the configuration items that needs changes are identified. The baseline versions of the configuration items are copied from the project data base.
- The changes are then included in the copied version of the item. The changes are subject to review by a designated team before testing and other quality assurance activity. Once the changes are approved, a new version is generated for distribution.
- Change request, change report and engineering change order are generated as part of change control activity. These documents are inprinted or electronic forms.

- Software Change Request Format :

1.0 Change request identification

1.1 Name, identification and description of configuration item: The name, version numbers of the software configuration is provided. Also brief description of configuration item is provided.

1.2 Requester and contact details: Name of the person requesting the change and contact details

1.3 Date, location, and time when change is requested.

2.0 Description of the change

2.1 Description: This section specifies a detailed description of the change request.

2.1.1 Background information, background information of the request.

2.1.2 Examples: supporting information, examples, error report and screenshoots.

2.1.3 The change: A detailed discussion of the change requested.

2.2 Justification for the change: detailed justification for the request.

2.3 Priority: The priority of the change.

- Software Change Report Format :

1.0 Change report identification

1.1 Name, identification and description of configuration item: The name, version numbers of the software configuration is provided. Also brief description of configuration item is provided.

1.2 Requester: the name and contact details of the person requesting the change.

1.3 Evaluator: The name of the person or team who evaluated the change request.

1.4 Date and time: when change report was generated.

2.0 Overview of changes required to accommodate request

2.1 Description of software configuration item that will be affected.

2.2 Change categorization: type of change

2.3 Scope of the change: The evaluator's assessment of the change.

2.3.1 Technical work required including tools required etc. A description of the work required to accomplish the change including required tools or other special resources are specified here

2.3.2 Technical risks: The risks associated with making the change are described.

3.0 Cost Assessment : Cost assessment of the requested change including an estimate of time needed.

4.0 Recommendation

4.1 Evaluator's recommendation: This section presents the evaluator's recommendation regarding the change.

4.2 Internal priority: how important is the change and priority assigned by the evaluator.

- Engineering Change Order Format

1.0 Change report identification

1.1 Name, identification and description of configuration item: The name, version numbers of the software configuration is provided. Also brief description of configuration item is provided.

1.2 Name of Requestor

1.3 Name of evaluator

2.0 Description of the change to be made

2.1 Description of software configuration that is affected.

2.2 Scope of the change required

The evaluator's assessment of the change in the configuration item.

2.2.1 Technical work and tools required: A description of the work and tools required to accomplish the change.

2.3 Technical risks: The risks associated with making the change are described in this section.

3.0 Testing and Validation requirements

A description of the testing and review approach required to ensure that the change has been made without any undesirable side effects.

3.1 Review plan: Description of reviews that will be conducted.

3.2 Test plan

Description of the test plans and new tests that are required.

- Benefits of change control management :
- It helps the developer identify the responsibility of code for which a developer is responsible.
- It provides a road map to the development process and encourages the developers to be more involved in their work.
- Version control mechanism helps the software tester to track the previous version of the product, thereby giving emphasis on testing of the changes.
- It helps the developer and tester to simultaneously work on multiple versions of the same product and still avoid any conflict and overlapping of activity.
- Its used by the managers to keep a control on the changes to the product thereby tracking and monitoring every change.

- It reassures the management.
- It provides a professional approach to control software changes.
- It also gives confidence to the customer regarding the quality of the product.

Auditing and Reporting

- Auditing :
- Auditing and reporting helps change management process to ensure whether the changes have been properly implemented or not, whether it has undesired effect on other components.
- A FTR concentrates on technical correctness of the changes to the configuration item whereas software configuration audit checks the parameters which are not checked in a FTR.
- Checklist for software audit :
  - ❑ Whether a FTR is carried out to check the technical accuracy of the changes made?
  - ❑ Whether the changes identified and reported in the change order have been incorporated.
  - ❑ Have the changes been properly documented in the configuration items?

- ❑ Whether the standards have been followed
- ❑ Whether the procedure for identifying, recording and reporting changes has been followed.

- As auditing is a formal process, its desirable to conduct the audit by another team than the team responsible for including the changes.
- Reporting :
- Its aka status accounting.
- It records all changes that lead to a new version.
- It's the bookkeeping of each release.
- The report includes:

- The report includes:
  - The changes included
  - The person responsible for the change.
  - The date and time of change.
  - The effect of change.
  - Reason for change.
- Every time a change is made it is assigned a unique number to identify it from the old previous version.
- It's very important in a situation where a large number of developers work on the same product at the same time and have little idea on other's work.
- Example: Page 55, block 2

- Example of reporting :

For example, in source code, reporting the changes may be as below:

```
**********************************************************************
**********************************************************************
**********************************************************************

# Title       : Sub routine Insert to Employee Data
# Version     : Ver 1.1.3
# Purpose     : To insert employee data in the master file
# Author      : John Wright
# Date        : 23/10/2001
# Auditor     : J  Waltson
# Modification History:
        12/12/2002 : by D K N
        To fix bugs discovered in the first release
        4/5/2003 : by S K G
        to allow validation in date of birth data
        6/6/2004 : by S S P
        To add error checking module as requested by the customer
```

FTR Vs Auditing

- FTR is a formal process to evaluate the technical accuracy of any process or changes.
- Audit is carried out by a separate team to ensure that proper change management procedure has been followed to incorporate the changes.