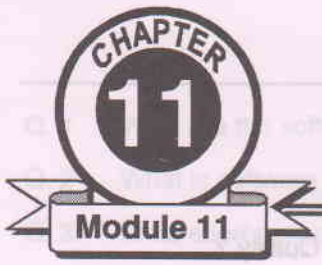

Change Management

Syllabus :

Software Configuration Management, Elements of SCM, SCM Process, Change Control.

11.1 Software Configuration Management

- Basically the output of any software process contains the information based on various inputs. These output can be broadly divided into three important categories as follow :
  - The computer programs
  - The work products and
  - The data that consist of the information produced.
- All these information parts collectively called as software configuration.
- If each configuration item simply led to other items, little confusion would result. During the process, change takes place which may be unfortunate. But change can occur any time and for any undefined reason. The change is the only constant.
- The first law of system engineering says that "No matter where you are in the system life cycle, the system will change, and the desire to change it will persist throughout the life cycle".
- **Following are four basic sources of change :**
  - Every new business conditions and market conditions may cause the change in product requirement and business rules definitions.

- Customers may require change in data presented by the information system. The customers may also demand various functionality delivered by their product of interest. The customers seek some new services. All these changes are customer oriented.
- As the business grows or its downsizing may cause various changes in priorities of the project and also restructuring of the team happens.
- The budget and time scheduling constraints are also important factors for change in system or product.

- SCM is a set of activities used for managing the change during the life cycle of computer software. The software configuration management can also be considered as a software quality assurance activity during the development process.

11.2 Elements of SCM

11.2.1 SCM Basics (Configuration Management System Elements)

Four basic elements that should exist when a configuration management system is developed :

1. **Component elements :** The tools in the file management system uses the software configuration item.
2. **Process elements :** The process elements or the procedures uses effective approach towards the change management in engineering and use of computer software.
3. **Construction elements :** The automated tools are used in construction or the development process and ensuring the validated components should be assembled.
4. **Human elements :** In order to make the effective use of SCM, the team makes the use of various tools and process feature.

These elements are not mutually exclusive. For example, component elements work in conjunction with construction elements as the software process evolves. Process elements guide many human activities that are related to SCM and might therefore be considered human elements as well.

11.2.2 Baselines

MU - Dec. 14

- The change is the only constant in software development life cycle. The customer want to modify the requirement as the model gets ready. Since in the beginning, even customer is not fully aware of the product requirement. As the development begins, customers need lot of changes in the requirement.

- Once customers modify their requirement, it is now manager who modifies the project strategy.
- Actually as time passes, all the people involved in the product development process come to know exact need, the approach and how it will be done in the prescribed amount of time.
- The additional knowledge is required to know the exact requirement. It is very difficult for most of the software engineers to accept this statement that most of the changes are justified.
- The baseline is SCM that help in development process without affecting much the schedule and control the changes.
- The IEEE provides a baseline as follows :

  "A specification and requirement that is agreed upon between customer and developer is a basis for the product development and these requirements can be changed only through change control procedures".

11.3 Software Configuration Items

MU - Dec. 14

- Basically SCI i.e. software configuration item is the integral part of software engineering development process. It is a part of large specification or we can say that one test case among large suite of test cases.
- In fact the SCI is a document or the program component like C++ functions or a Java applet.
- Most of the organizations use software tools under configuration control to help development process.
- In fact the editors, compilers, browsers and various automated tools are the integral part of software configuration.
- In fact the SCIs are catalogued in the project database with a single name and they form configuration objects.
- These objects are configuration object and it has a name, attribute and it has certain relationship with other configuration objects.

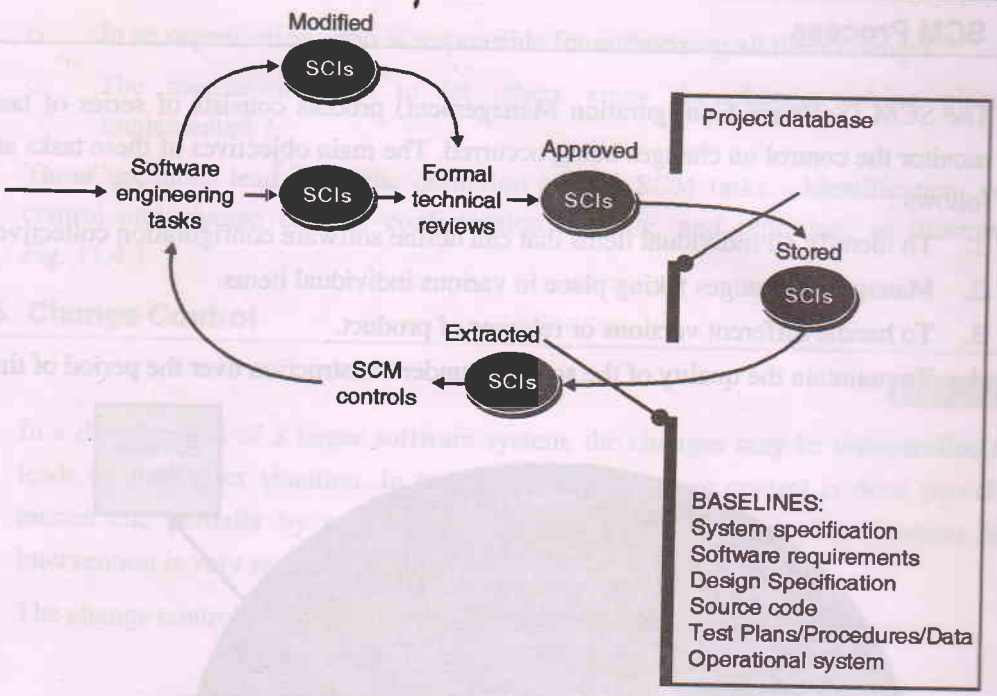

Fig. 11.3.1 : Baselined SCIs and the project database

- Referring to Fig. 11.3.2, the configuration objects, Design Specification, Data Model, Component N, Source Code and Test Specification are each defined separately.

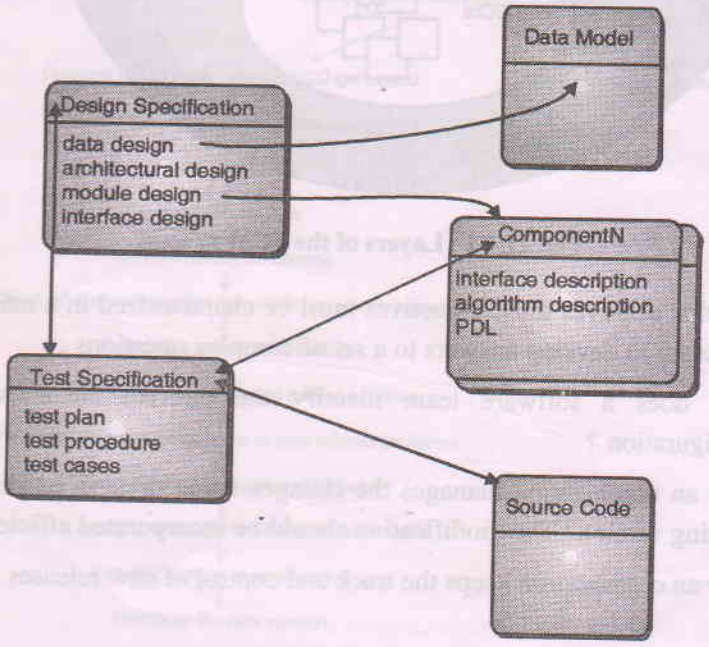

Fig. 11.3.2 : Configuration objects

11.4 SCM Process

- The SCM (Software Configuration Management) process consists of series of task to monitor the control on changes being occurred. The main objectives of these tasks are as follows :
  1. To identify all individual items that can define software configuration collectively.
  2. Manage the changes taking place in various individual items.
  3. To handle different versions or releases of product.
  4. To maintain the quality of the software under construction over the period of time.

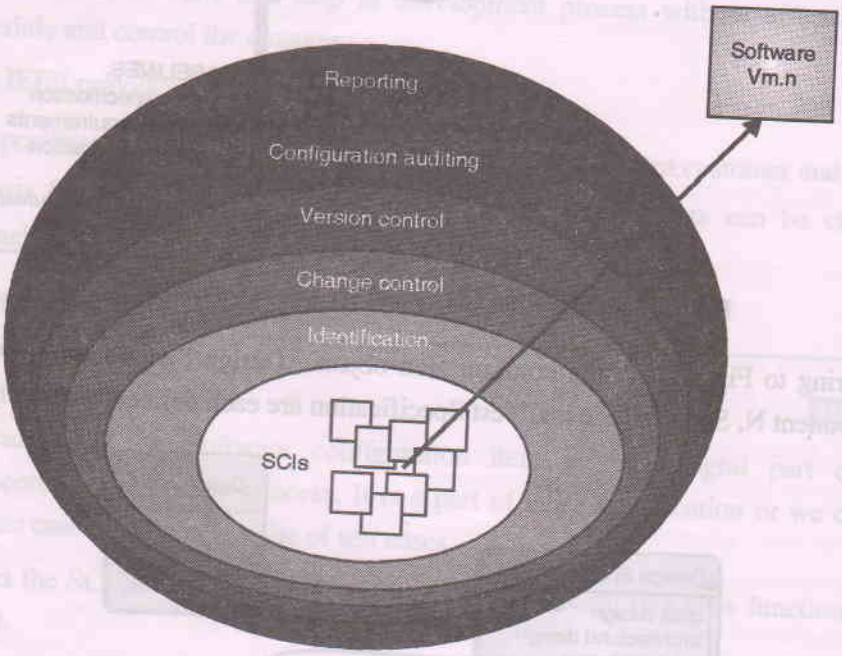

Fig. 11.4.1 : Layers of the SCM process

- A process that achieves these objectives must be characterized in a manner that enables a software team to develop answers to a set of complex questions :
  - How does a software team identify the discrete elements of a software configuration ?
  - How an organization manages the changes being done in existing release or the existing version? The modification should be incorporated efficiently.
  - How an organization keeps the track and control of new releases ?

- In an organization, who is responsible for authorizing all these changes ?
- The mechanism used to let others know the changes taking place and implemented ?

- These questions lead us to the definition of five SCM tasks - identification, version control and change control, configuration auditing, and reporting, as illustrated in Fig. 11.4.1.

11.5 Change Control

MU - Dec. 14

- In a development of a larger software system, the changes may be uncontrolled and it leads to a complex situation. In such projects the change control is done partially by human and partially by some automated tools. In such a complex situation human intervention is very much necessary.
- The change control process is explained in the following Fig. 11.5.1.

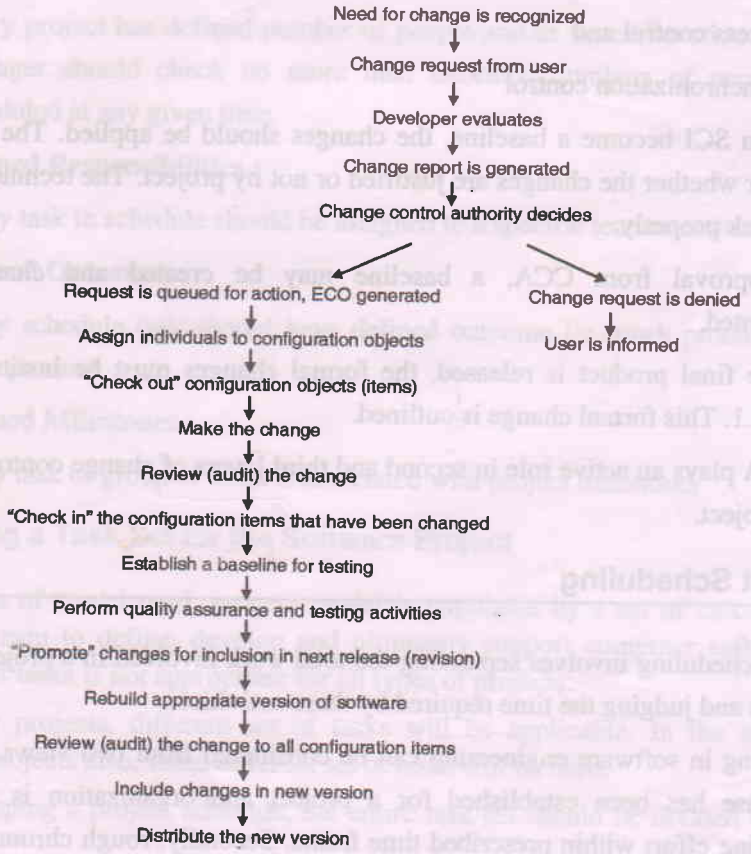

Fig. 11.5.1 : The change control process

- The change request is first submitted and then evaluated by a technical support staff by taking into consideration its potential side effects and the overall impact on other objects in the product. The other parameters to be evaluated are system functions, the cost of project etc.
- Based on the result of the evaluation treated as a change report, the implementation is taken into consideration. This report is submitted by change control authority i.e. CCA. The CCA is a person or a group who has the final authority to take decision on any changes and their priority.
- After approval from CCA, a change order called ECO (engineering change order) is generated for each of the change.
- The object to be changed can be placed in a directory that is controlled solely by the software engineer making the change.
- These version control mechanisms, integrated within the change control process, implement two important elements of change management :
  - access control and
  - Synchronization control
- Before an SCI become a baseline, the changes should be applied. The developer will look after whether the changes are justified or not by project. The technical requirement must check properly.
- After approval from CCA, a baseline may be created and change control is implemented.
- Once the final product is released, the formal changes must be instituted as per the Fig. 11.5.1. This formal change is outlined.
- The CCA plays an active role in second and third layers of change control based on size of the project.

11.6 Project Scheduling

- Project scheduling involves separating the total work involved in a project into separate activities and judging the time required by these activities.
- Scheduling in software engineering can be considered from two views. First, end date for release has been established for a project and organization is responsible for distributing effort within prescribed time frame. Secondly, rough chronological work are discussed and end date is set by organisation.

- Thus, project scheduling is an activity that distributes estimated effort across the planned project duration by specific allocation of effort to the specific task in software engineering.
- Basic principles for software project scheduling are :

1. **Compartmentalization :**

   Project is divided or compartmented into number of manageable activities, actions and tasks.

2. **Interdependency :**

   Interdependency between each compartmentalized activity, action task must be determined.

3. **Time allocation :**

   Each task is assigned some work unit i.e. effort by person in days.

4. **Effort Validation :**

   Every project has defined number of people and as time allocation occurs, project manager should check no more than allocated numbers of people have been scheduled at any given time.

5. **Defined Responsibilities :**

   Every task in schedule should be assigned to a specific team member.

6. **Defined Outcomes :**

   Every schedule task should have defined outcome i.e. work product or a part of work product.

7. **Defined Milestones :**

   Every task or group of tasks is associated with project milestone.

11.6.1 Defining a Task Set for the Software Project

- Regardless of model used, process model is populated by a set of tasks that enable a software team to define, develop and ultimately support computer software. But the same set of tasks is not appropriate for all types of projects.
- For larger projects, different set of tasks will be applicable. In the same way, for complex projects also, some different set of tasks will be used.
- For developing a project schedule, the entire task set should be divided on the project time line.

- For each project, the set of tasks will vary depending on the type of the project.
- Following are some types of projects :
  - Concept development projects
  - New project having certain application
  - Application enhancement projects
  - Maintenance projects
  - Reengineering projects
- In a particular type of project, many factors influence the task set to be chosen.
- These factors include :
  - Size of the project
  - Number of users
  - Stability requirements
  - User friendliness
  - Ease of communication between the application developer or user
  - Performance
  - Technology used

An example set of tasks :

- Consider the example of concept development projects. Following are the sets of tasks those can be applied :
  - Concept scope
  - Concept planning
  - Risk assessment and management
  - Proof of concept
  - Implementation
  - Customer feedback and reaction
- These are certain activities or the set of tasks those can be applied for the concept development projects.

11.6.2 Scheduling

MU - Dec. 14

- Scheduling of different engineering activities can use the available project scheduling tools and techniques.
- Following are some project scheduling tools and techniques :
  - PERT (Program Evaluation and Review Technique)
  - CPM (Critical Path Method)
- These are two project scheduling methods that can be applied to the software development process.
- Both tools use the data and information received from the earlier developments.
- Both tools allow to determine the critical path, duration of the projects, and time estimate for the individual activity, efforts taken, duration.

Time-line charts :

- A time-line chart, also called Gantt chart, is generated on the basis of the start date inputs for each task.
- Fig. 11.6.1 shows an example Gantt chart. It elaborates the project schedule.
- In the left hand column of the Fig. 11.6.1, all the project tasks are listed. The horizontal lines indicate the duration of the task.
- For concurrent activities, multiple horizontal lines are used.

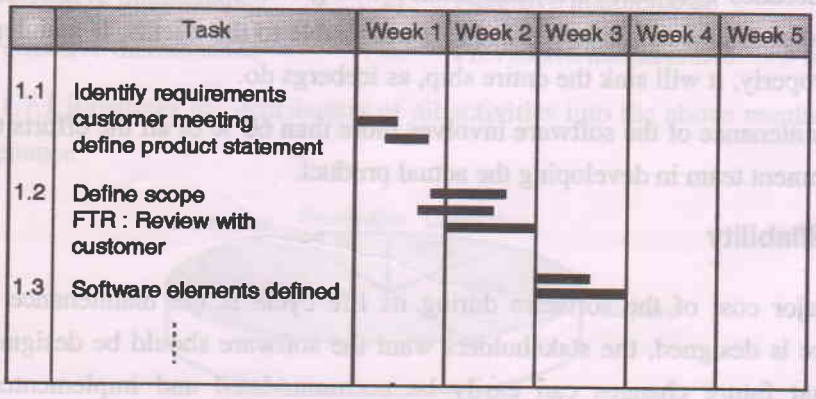

Fig. 11.6.1 : An example time-line chart

MU - Dec. 14

11.6.3 Tracking the Schedule

- The project schedule is nothing but a road map that defines the tasks and the milestones to be tracked.
- Tracking is achieved in different ways as follows :
  - Conduct periodic meeting to find out the progress status and problems.
  - Evaluate the process.
  - Check whether all the activities are completed within the deadline or by the schedule date.
  - Compare the planned date and actual start date for each activity.
  - Meet practitioners informally to assess the progress.
- All these tracking techniques are used by the project managers.

11.7 Software Maintenance

- The software maintenance is an activity that actually begins after the software product delivered at the client's end. In software maintenance, the modifications are carried out or the updates in the software are taken place.
- In software maintenance phase no major changes are implemented.
- In software maintenance, the changes are done in the existing program or the some small new functionality is added.
- Three decades ago, software maintenance was given the name as **iceberg**, where the potential problems reside inside and it is not visible to the clients. If maintenance is not done properly, it will sink the entire ship, as icebergs do.
- The maintenance of the software involves more than 60 % of all the efforts taken by the development team in developing the actual product.

11.7.1 Modifiability

- The major cost of the software during its life cycle is the maintenance cost. When software is designed, the stakeholders want the software should be designed in such a way that future changes can easily be accommodated and implemented with less maintenance cost. If implementation is easy, then the cost of maintenance will automatically be decreased.