Software Engineering

SOFTWARE PROJECT PLANNING - IV

Software Configuration Management

Gagan Deep
Founder & Director
Rozy Computech Services
Kurukshetra, rozygag@yahoo.com
www.rozyph.com

SOFTWARE CONFIGURATION MANAGEMENT

- When we develop a software, the product (software) undergoes many changes in their maintenance phase, we need to handle these changes effectively.
- Several individual (programs) works together to achieve this common goal. These individual produces several work product (SC Items) e.g. intermediate version of modules or test data used during debugging, part of final product.
- The item that comprise all information produced as part of the software process are collectively called a software configuration.

- As software development progresses, the number of Software Configuration Items (SCI’s) increases rapidly.
- These are managed and controlled by SCM. This is where we need software configuration management.
- *A configuration of product refers not only to the product's constituent, but also to specific version of the component.*
- Therefore SCM is the discipline which
  - Identify change
  - Monitor and control change
  - Ensure the proper implementation of change made to item.
  - Auditing and reporting on the change made.

- Configuration management(CM) is an art of identifying, organising and controlling modifications to software being built by a programming team.
- The goal is to maximize productivity by minimizing mistakes(errors).
- CM is essential due to the inventory management, library management and updation management of the items important for the project.

Importance of SCM

- It is useful in controlling and managing the access to various SCIs e.g. by preventing the two member of a team for checking out the same component for modification at same time.
- It provide the tool to ensure that changes is being properly implemented
- It has the capability of defining and storing the various constituent of a software
- SCM is used in keeping a system in a consistent state by automatically generating derived version upon modification of same component.

SCM Process

- It uses the tools which ensure that the required change has been properly implemented to the appropriate component (item). Thus it answer to the following questions
  - How and when the change will be done to software?
  - How to ensure the changes are done correctly?
  - What procedure will be adopted to make the change?
  - Who will be responsible for change ?
  - How to rank/order for importance of various change or items?
  - How to manage the many version of the program?

- These questions lead us to the definition of five SCM task:
  - Identification
  - Version Control
  - Change Control,
  - Configuration Auditing, and
  - Status Reporting.

- Identification
  - Basic Object : Unit of Text created by a software engineer during analysis, design, code, or test.
  - Aggregate Object : collection of basic objects and other aggregate objects. Design Specification is an aggregate object.
  - Each object has a set of distinct features that identify it uniquely: a name, a description, a list of resources, and a "realization".
  - The interrelationships between configuration objects can be represented with a *Module Interconnection Language*(MIL).

- Version Control
  - *Version Control* combines *procedures and tools to manage different versions of configuration objects that are created during the software process.*
  - Clemm describes version control in the context of SCM: *Configuration management allows a user to specify alternative configurations of the software system through the selection of appropriate versions. This is supported by associating attributes with each software version, and then allowing a configuration to be specified [and constructed] by describing the set of desired attributes.*

- Change Control
- James Bach describes change control in context of SCM is: *Change control is vital. But the forces that make it necessary* also make it annoying.
- *We worry about change because a small confusion in the code can create a big failure in the product. But it can also fix a big failure or enable wonderful new capabilities.*
- *We worry about change because a single rogue developer could sink the project; yet brilliant ideas originate in the minds of those rogues, and*
- *A burdensome change control process could effectively discourage them from doing creative work.*

Bach recognizes that we face a balancing act.

- Too much change control and we create problems.
- Too little, and we create other problems.
- A *Change request* is submitted and evaluated to assess technical merit, potential side effects, overall impact on other configuration objects and system functions, and projected cost of the change.
- The results of the evaluation are presented as a *change report*, which is used by a *change control authority(CCA)* - a person or group who makes a final decision on the status and priority of the change.

- The "check-in" and "check-out" process implements two important elements of change control - access control and synchronization control.
- *Access control* governs which software engineers have the authority to access and modify a particular configuration object.
- *Synchronization control* helps to ensure that parallel changes, performed by two different people, don't overwrite one another.
- The CCA may be composed of one person - the project manager - or a number of people (e.g., representatives from software, hardware, database engineering, support, marketing).

The role of the CCA is to take a global view, that is, to assess the impact of change beyond the SCI in question.

- How will the change affect hardware?
- How will the change affect performance?
- How will the change modify customer's perception of the product?
- How will the change affect product quality and reliability?
- These and many other questions are addressed by the CCA.

Configuration Auditing

- How can we ensure that the change has been properly implemented? The answer is two fold:
  - Formal technical reviews and
  - The software configuration audit.
- The Formal Technical Review focuses on the technical correctness of the configuration object that has been modified. The reviewers assess the SCI to determine consistency with other SCIs, omissions, or potential side effects. A formal technical review should be conducted for all but the most trivial changes.
- A Software Configuration Audit complements the formal technical review by assessing a configuration object for characteristics that are generally not considered during review.

- **Status Reporting**
- *Configuration status reporting* (sometimes called *status accounting*) is an SCM task that answers the following questions:
  - What happened?
  - Who did it?
  - When did it happen?
  - What else will be affected?
- Status reporting provides information about each change to those with a need to know.

Thanks!

Gagan Deep
rozygag@yahoo.com
www.rozyph.com