Other Software Processes

Other Processes

- Development Process is the central process around which others revolve
- Methods for other processes often influenced by the dev process
- We have looked at various models for dev process; a real process may be derived from a model

Other Processes

- Project management process
- Inspection process
- Configuration management process
- Change management process
- Process Management Process

- Will briefly look at these now

Project Management Process

Background

- Development process divides development into phases and activities
- To execute it efficiently, must allocate resources, manage them, monitor progress, take corrective actions, …
- These are all part of the PM process
- Hence, PM process is an essential part of executing a project

PM Process Phases

- There are three broad phases
  - Planning
  - Monitoring and control
  - Termination analysis
- Planning is a key activity that produces a plan, which forms the basis of monitoring

Planning

- Done before project begins
- Key tasks
  - Cost and schedule estimation
  - Staffing
  - Monitoring and risk mgmt plans
  - Quality assurance plans
  - Etc.
- Will discuss planning in detail later

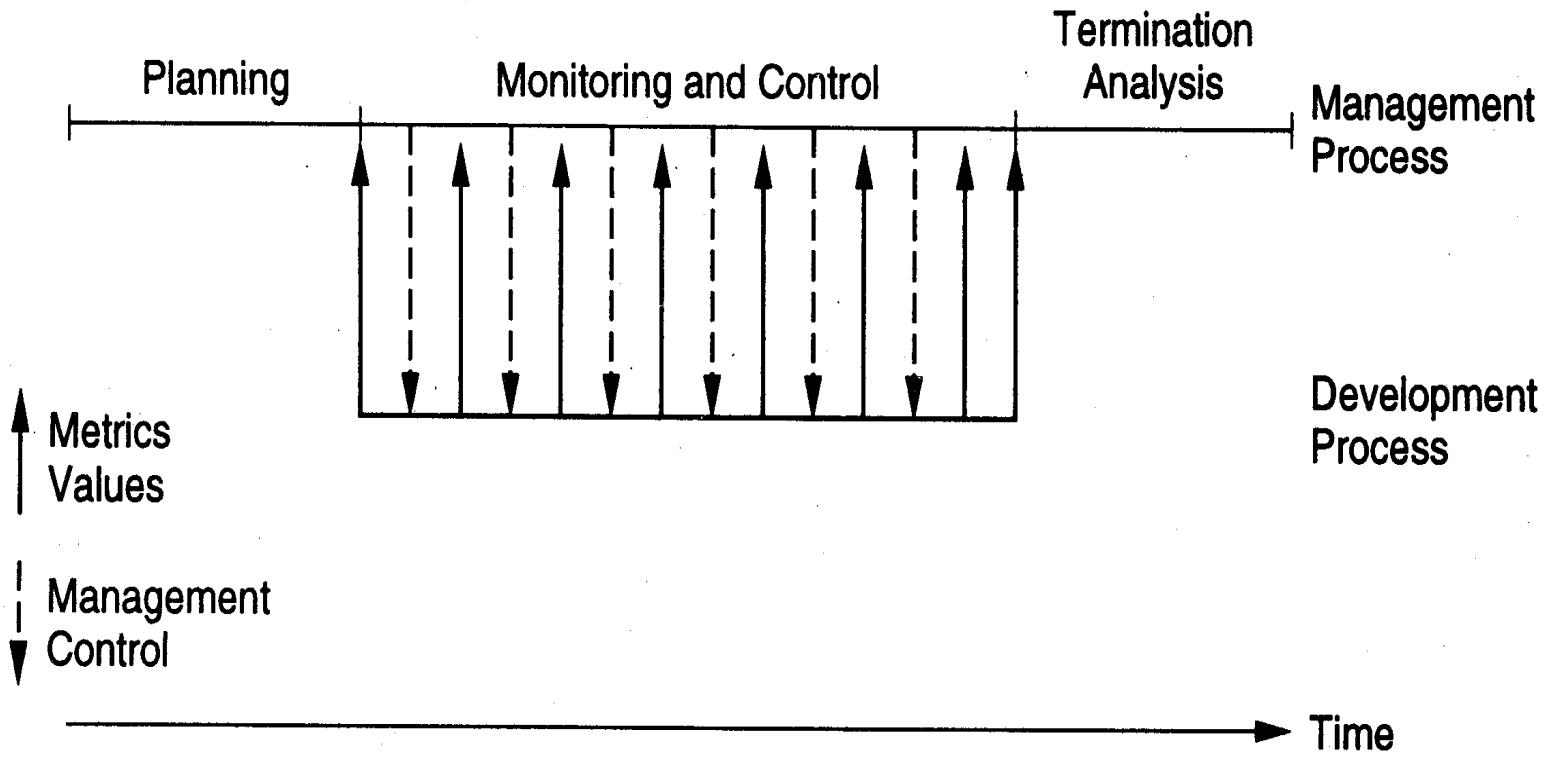
Monitoring and control

- Lasts for the duration of the project and covers the development process
  - Monitors all key parameters like cost, schedule, risks
  - Takes corrective actions when needed
  - Needs information on the dev process – provided by metrics

Termination Analysis

- Termination analysis is performed when the development process is over
- Basic purpose: to analyze the perf of the process, and identify lessons learned
- Also called postmortem analysis

Relationship with Dev Process



The Inspection Process

Background

- Main goal of inspection process is to detect defects in work products
- First proposed by Fagan in 70s
- Earlier used for code, now used for all types of work products
- Is recognized as an industry best practice
- Data suggests that it improves both Q&P

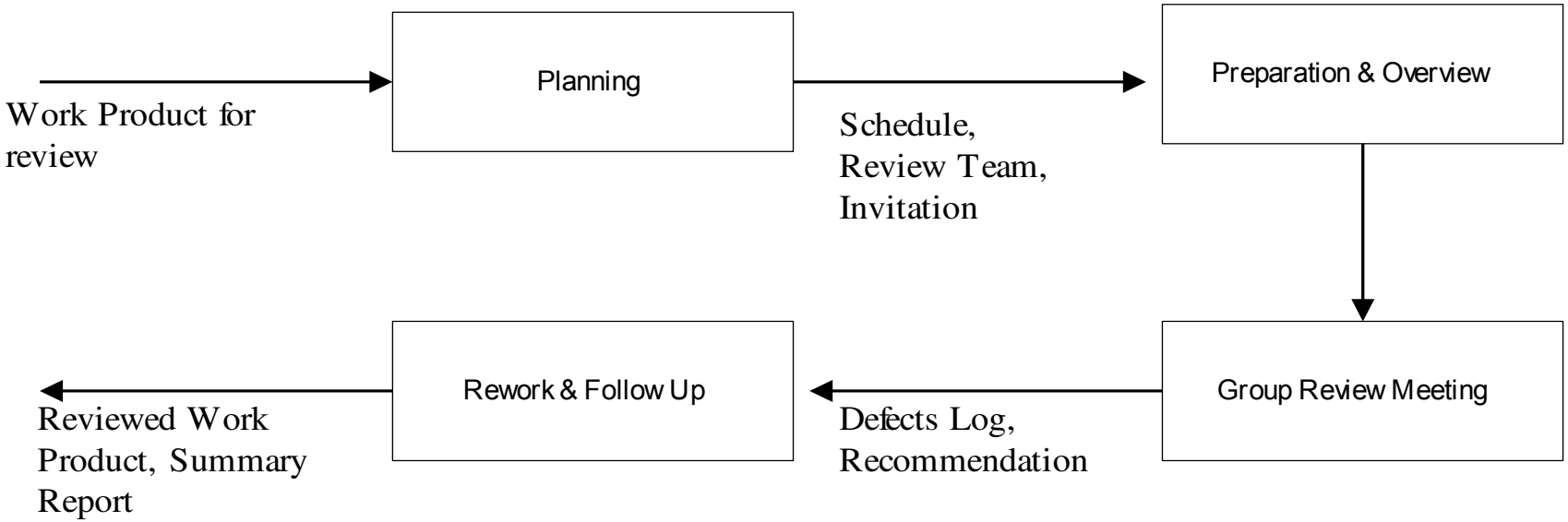
Background

- Defects injected in sw at any stage
- Hence must remove them at every stage
- Inspections can be done on any document including design docs and plans
- Is a good method for early phases like requirements and design
- Also useful for plans (PM plans, CM plans, testing plans,…)

Some Characteristics

- Conducted by group of technical people for technical people (i.e. review done by peers)
- Is a structured process with defined roles for the participants
- The focus is on identifying problems, not resolving them
- Review data is recorded and used for monitoring the effectiveness

A Review Process



Planning

- Select the group review team – three to five people group is best
- Identify the moderator – has the main responsibility for the inspection
- Prepare package for distribution – work product for review plus supporting docs
- Package should be complete for review

Overview and Self-Review

- A brief meeting – deliver package, explain purpose of the review, intro,…
- All team members then individually review the work product
  - Lists the issues/problems they find in the self-preparation log
  - Checklists, guidelines are used
- Ideally, should be done in one sitting and issues recorded in a log

Self-Review Log

Project name:

Work product name and ID:

Reviewer Name

Effort spent (hours)

Defect list

No	Location	Description	Criticality

Group Review Meeting

- Purpose – define the final defect list
- Entry criteria – each member has done a proper self-review (logs are reviewed)
- Group review meeting
  - A reviewer goes over the product line by line
  - At any line, all issues are raised
  - Discussion follows to identify if a defect
  - Decision recorded (by the scribe)

Group Review Meeting…

- At the end of the meeting
  - Scribe presents the list of defects/issues
  - If few defects, the work product is accepted; else it might be asked for another review
  - Group does not propose solutions – though some suggestions may be recorded
  - A summary of the inspections is prepared – useful for evaluating effectiveness

Group Review Meeting…

- Moderator is in-charge of the meeting and plays a central role
  - Ensures that focus is on defect detection and solutions are not discussed/proposed
  - Work product is reviewed, not the author of the work product
  - Amicable/orderly execution of the meeting
  - Uses summary report to analyze the overall effectiveness of the review

Summary Report Example

Project	XXXX
Work Product Type	Project plan
Size of work product	14 pages
Review team	P1, P2, P3
Effort (person hours)	
Preparation	10 (total)
Group meeting	10
Total	20

Summary Contd.

Defects	
No of critical defects	0
No of major defects	3
No of minor defects	16
Total	19
Review status	Accepted
Reco for next phase	Nil
Comments	Nice plan

Rework and Follow Up

- Defects in the defects list are fixed later by the author
- Once fixed, author gets it OKed by the moderator, or goes for another review
- Once all defects/issues are satisfactorily addressed, review is completed and collected data is submitted

Roles and Responsibilities

- Main roles: Moderator, reader, scribe, author, reviewer
- Moderator – overall responsibility
- Reviewer – to identify defects
- Reader – not there in some processes, reads line by line to keep focus
- Scribe – records the issues raised

Guidelines for Work Products

Work Product	Inspection focus	Participants
Req Spec	Meet customer needs Are implementable Ommissions, inconsistencies, ambiguities	Customer Designer Tester, Dev Analyst
HLD	Design implements req Design is implementable Ommissions, quality of design	Req author Designer Developer

Guidelines for Work Products

Code	Code implements design Code is complete and correct Defects in code Other quality issues	Designer Tester Developer
Test cases	Set of test cases test all SRS conditions Test cases are executable Are perf and load tests there	Req author Tester Proj mgr
Proj Mgmt Plan	Plan is complete and specifies all components of the plan Is implementable Omissions and ambiguities	Proj mgr Another Proj mgr SEPG member

Summary

- Purpose of reviews: to detect defects
- Structured reviews are very effective - can detect most of the injected defects
- For effective review, process has to be properly defined and followed
- Data must be collected and analyzed

Configuration Management Process

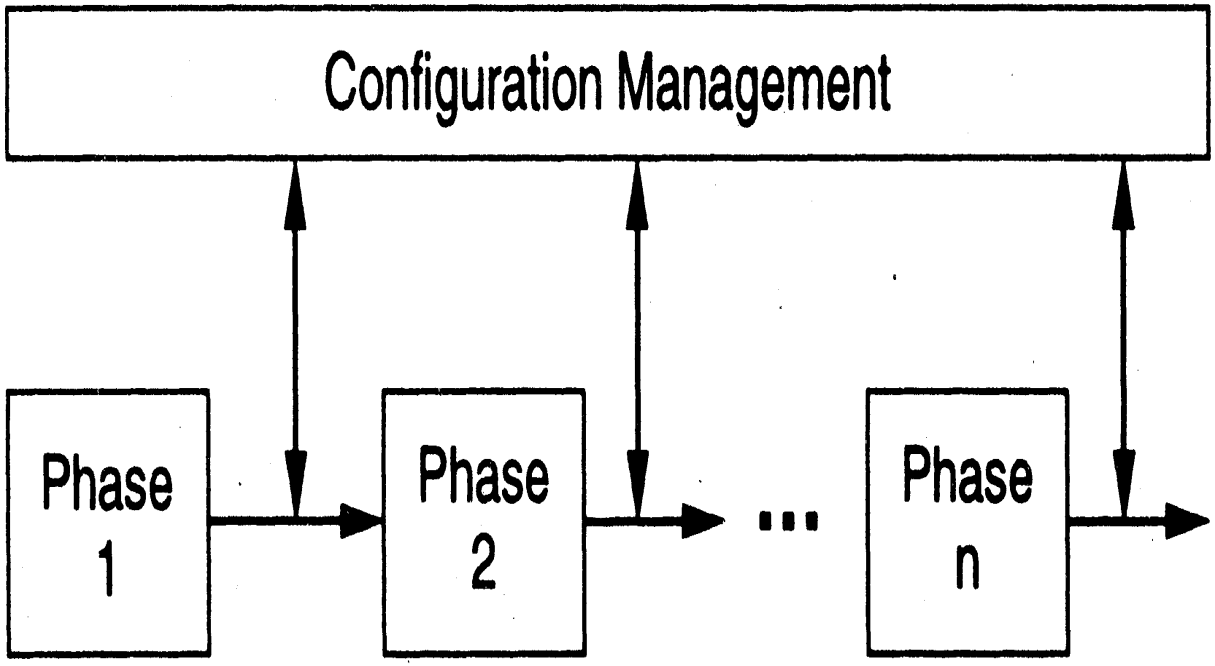
Background

- A software project produces many items - programs, documents, data, manuals, …
- All of these can be changed easily – need to keep track state of items
- SCM: Systematically control the changes
  - Focus on changes during normal evolution; req changes will be handled separately
- CM requires discipline as well as tools

Background

- SCM often independent of dev process
  - Dev process looks at macro picture, but not on changes to individual items/files
- As items are produced during the dev process, it is brought under SCM
- SCM controls only the products of the development process

SCM Process and Dev process



Need for CM

- To satisfy the basic project objective of delivering the product to the client
  - What files should comprise the product
  - What versions of these files
  - How to combine these to make the product
- Just for this, versioning is needed, and state of diff items has to be tracked
- There are other functions of CM also

Functionality Needed

- Give states of programs
- Give latest version of a program
- Undo a change and revert back to a specified version
- Prevent unauthorized changes
- Gather all sources, documents, and other information for the current system

CM Mechanisms

- Configuration identification and baselining
- Version control
- Access control

- These are the main mechanisms, there are others like naming conventions, directory structure,…

Configuration Items

- Sw consists of many items (some artifact)
- In CM some identified items are placed under CM control
- Changes to these are then tracked
- Periodically, system is built using these items and baselines are established
- Baseline – logical state of the system and all its items; is a reference point

Version and access control

- Key issues in CM
- Done primarily on source code through source code control systems, which also provide access control
- Allows older versions to be preserved and hence can undo changes
- Eg: CVS and VSS are commonly used; Clear case for large projects

Version and Access Control

- When programmer developing code – is in private area
- When code is made available to others, it goes in an access-controlled library
- For making changes to an item in library, it has to be checked out
- Changes made by checking-in the item – versioning is automatically done
- Final system is built from the library

Version/Access Control

- Generally both version and access control done through CM tools like VSS, SCCS
- Tools limit access to specified people - formal check in, check out procedures
- Automatic versioning done when a changed file is checked-in
- Check-in, check-out control may be restricted to a few people in a project

CM Process

- Defines the activities for controlling changes
- Main phases
  - CM Planning
  - Executing the CM process
  - CM audits

CM Planning

- Identify items to be placed under CM
- Define library structure for CM
- Define change control procedures
- Define access control, baselining, reconciliation, procedures
- Define release procedure

CM Audit

- During project execution CM procedures have to be followed (e.g. moving items between directories, naming, following change procedures, …)
- Process audit has to be done
- CM audit can also check if items are where they should be

Summary – CM

- CM is about managing the different items in the product, and changes in them
- Developing a CM plan at the start is the key to successful to CM
- CM procedures have to be followed; audits have to be performed
- Requires discipline and tools

Requirements Change Management Process

Background

- Requirements change. At any time during the development
- Changes impact the work products and the various configuration items
- Uncontrolled changes can have a huge adverse impact on project in cost/sched
- Changes have to be allowed, but in a controlled manner

A Change Mgmt Process

- Log the changes
- Perform impact analysis on the work products and items
- Estimate impact on effort and schedule
- Review impact with stakeholders
- Rework the work products/items

Changes

- Change often triggered by *change request*
- Change req log keeps a record of requests
- Impact analysis for a change request involves identifying the changes needed to diff items, and the nature of change
- The impact of changes on the project is reviewed to decide whether to go ahead
- Cumulative changes also often tracked

Process Management Process

Background

- A process is not a static entity – it has to change to improve to improve the Q&P
- Focus of process management is to evaluate and improve the process
- Is different from project management which focuses on a project
- Process management is an advanced topic

Software Process Improvement

- To improve the process, an org must understand the current process
  - Requires process be **properly documented**
  - Properly executed on projects
  - **Data** is collected from projects to understand the **performance** of process on projects
- Changes to process are best made in small increments

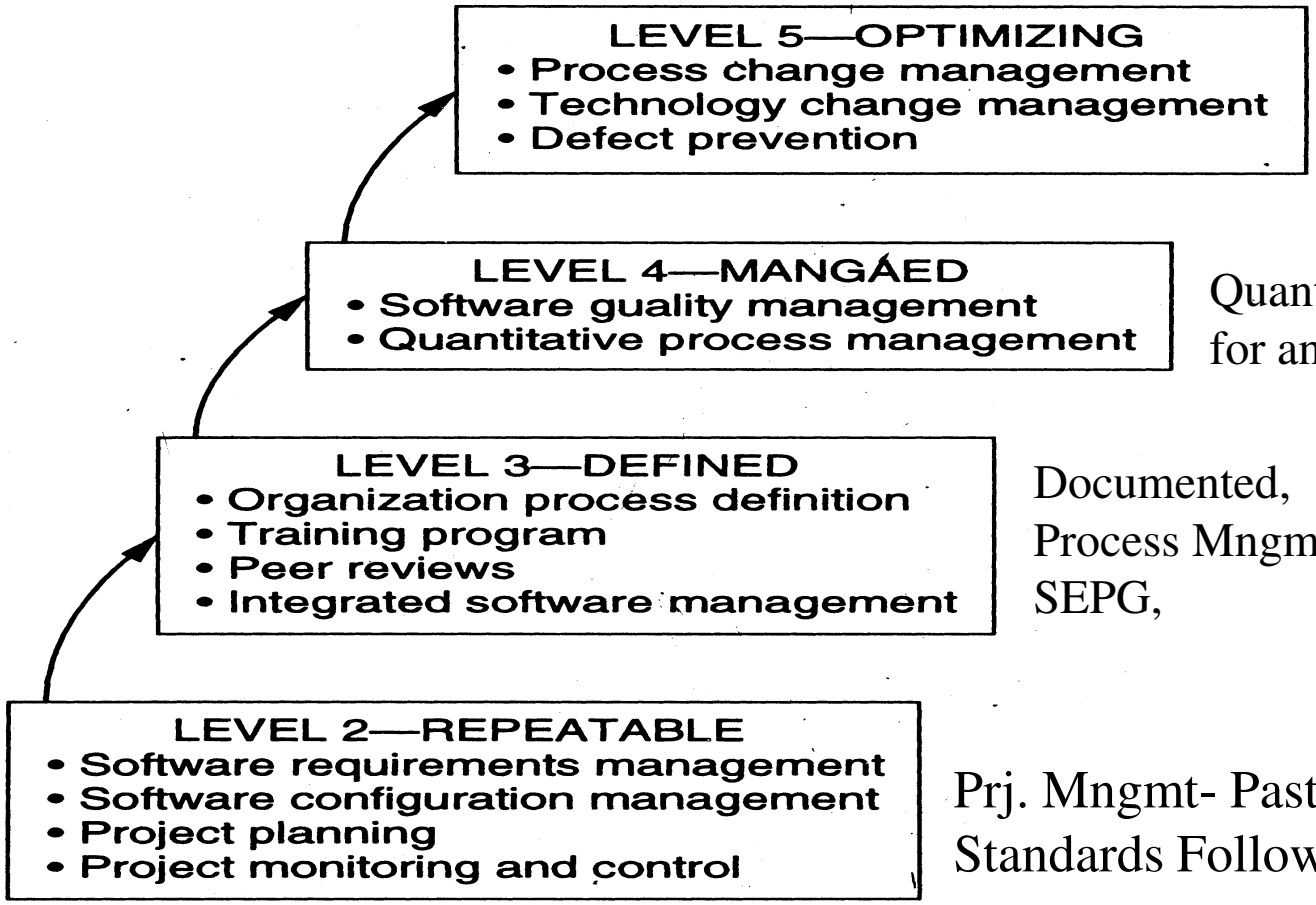
Software Process Improvement Frameworks

- What changes should be made to the process and when
- Frameworks suggest ways of how process improvement can proceed
- Capability Maturity Model (CMM) is one of the most common frameworks

CMM

- CMM has five maturity levels (***Well Defined***) for a software process (level 1 is ad-hoc)
- In a level, process has some capabilities (***expectations***) and lays the foundation for next level
- For moving from one level to another, CMM specifies areas to focus on
- Is used heavily by sw industry

CMM



Level 1- Personalized Approach