CALIFORNIA STATE UNIVERSITY
FULLERTON™

COLLEGE OF ENGINEERING
AND COMPUTER SCIENCE

Advanced Software Process

Part III: The Defined Process

9. Software Standards

Dave Garcia-Gomez
Faculty / Lecturer
Department of Computer Science

Course Roadmap

Part III: The Defined Process

- Once organizations have mastered the basic capabilities described in Part II, the performance of their projects will have sufficiently stabilized to permit orderly process improvement.
- The priority needs are then to improve from Level 2 to Level 3.
- This improvement will establish a more consistent and uniform process across the organization and provide a coherent framework for organized learning.

Part III: The Defined Process

- Part III describes the key topics that Level 2 organizations must address to advance to Level 3.
- A more detailed listing of the total set of improvement actions is included in Appendix A.

Part III: The Defined Process

- The subjects requiring priority management attention at this point (from 2 to 3) are:
  - Standards
  - Software inspections
  - Testing
  - Advanced configuration management topics
  - Process models and architecture
  - Software Engineering Process Group (SEPG)

Part III: The Defined Process

Ch.9 – Software Standards
- – Benefits of standards, establishment of standards program

Ch.10 – Software Inspections
- – What are software inspections, how an inspection program is initiated, inspection guidelines

Ch.11 – Software Testing
- – Principles of testing, methods, planning, management

Ch.12 – Software Configuration Management (II)
- – Ch.7 basics, management of requirements and design changes, configuration auditing

Ch.13 – Defining the Software Process
- – Process models, process architectures

Ch.14 – The SE Process Group
- – Considerations involved in establishing an SEPG

Software Standards

- Software Standards
- Definitions
- The Reasons for SW Standards
- Benefits of Standards
- Examples of Some Major Standards
- Establishing SW Standards
- Standards versus Guidelines

Software Standards

- A standard is a rule or basis for comparison that is used to assess the size, content, value, or quality of an object or activity.
- In software, two kinds of standards are used:
  - Describes the nature of the object to be produced
  - Defines the way the work is to be performed
- Other standards
  - Languages, coding conventions, commenting, change flagging, error reporting

Software Standards

- Procedures are closely related to standards.
- For example, there are standards for software reviews and audits as well as procedures for conducting them.
  - A review standard specifies review contents, preparatory materials, participants, responsibilities, and the resulting data and reports.
  - The procedure for conducting the review describes how the work is actually to be done, by whom, when, and what is done with the results.

Software Standards

- Representative Software Standards [Table 9.1]
  - Software quality assurance plans
  - Software development notebooks
  - Software development plans
  - Software reviews and audits
  - Software requirements
  - Software design documentation
  - Software test plans
  - Software quality assurance reviews
  - Software configuration management
  - Problem reporting/corrective action
  - Software documentation

Software Standards

- Representative Software Procedures [Table 9.2]
  - Auditing Software development notebooks
  - Reviewing a software development plan
  - Conducting software reviews
  - Conducting software audits
  - Reviewing software requirements
  - Reviewing software design documents
  - Reviewing software test plans
  - Auditing the software testing process
  - Conducting SQA reviews
  - Performing Software Configuration Management (SCM)
  - Auditing Software Configuration Management systems
  - Handling problem reporting/corrective action
  - Auditing problem reporting/corrective action systems
  - Reviewing software documentation

Software Standards

- There are many different ways to establish standards for an organization.
- TRW software development policies, procedures, and standards [Figure 9.1]
  - TRW has also published a guidebook of DoD regulations, specifications, and standards.
  - Although it is somewhat out of date, the guidebook does provide a helpful overview of a number of military software standards.

Software Standards

- TRW software development policies, procedures, and standards [Figure 9.1]

Definitions

- Before discussing standards and procedures, it is necessary to establish some definitions.
- Software terms can be grouped into categories, with the meaning understood in the textbook, as follows:

Definitions

- Authoritative direction on what is to be done:
  - Policy

    A governing principle, typically used as the basis for regulations, procedures, or standards, and generally stated by the highest authority in the organization
  - Regulation

    A rule, law, or instruction, typically established by some legislative or regulatory body
  - Specification

    The precise and verifiable description of the characteristics of a product

    A process specification define a method, procedure, or process to be used in performing a task.

    Specifications are produced by technical experts.

Definitions

- The characterization of how a task is to be performed or the required characteristics of the result:
  - Guideline

    A suggested practice, method, or procedures
  - Procedure

    A defined way to do something
  - Standard

    A rule or basis for comparison that is used to assess size, content, or value, typically established by common practice or by a designated standards body

Definitions

- The ways in which tasks are accomplished:
  - Convention

    A general agreement on practices, methods, or procedures
  - Method

    A regular, orderly procedure or process for performing a task
  - Practice

    A usual procedure or process, typically a matter of habit or tacit agreement
  - Process

    A defined way to perform some activity

Definitions

- The focus in this chapter is on:
  - Guidelines

    A suggested practice, method, or procedure, typically issued by some authority.
  - Procedures

    A defined way to do something, generally embodied in a procedures manual.
  - Standards

    A rule or basis for comparison that is used to assess size, content, or value, typically established by common practice or by a designated standards body.

The Reasons for Software Standards

- Standards are needed when many people, products, or tools must coexist.
- Standards are essential for establishing common support environments, performing integration, or conducting system test.
  - The fact that everyone knows and understands a common way of doing the same tasks makes it easier for the professionals to move between projects, reduces the need for training, and permits a uniform method for reviewing the work and its status.

The Reasons for Software Standards

- Standards also promote the consistent use of better tools and methods.
- When everyone uses common coding conventions and commenting guidelines, for example, it is practical for programmers to review each others’ work and easier for them to understand it.
- This both facilitates design and code inspections and improves the maintainability of the finished product.

Benefits of Standards

- While there is *little quantitative evidence* that supports the use of standards, most experienced software managers can cite at least one standard that was key to a program's success.
- While standards alone will <u>not</u> make the difference between project success and failure, they clearly help.

Benefits of Standards

- Thayer (1982) has surveyed SW managers to determine their views on the key software problems and their most effective solutions.
  - Of all the solutions listed, the use of enforcement of standards and procedures ranked first.

Examples of Some Major Standards

- IBM's Federal Systems Division (FSD) has established a family of standards for software development [Table 9.3 Software Design Practices].
- The standards have been an important contributor to the high quality and productivity of IBM's FSD programming group.
  - A set of code management standards

    Programming languages, coding standards and conventions, computer product support SW, hierarchical program control library, SW development environment
  - Conventions

    Rules for naming programs, designating variables, and for writing comments

Examples of Some Major Standards

- IBM's Federal Systems Division (FSD) Software Design Practices [Table 9.3]
  - Systematic programming practices
    - Logical expression
    - Program expression
    - Program design
    - Program design verification
  - Systematic design practices
    - Data design
    - Modular design
  - Advanced design practices
    - Software system specification
    - Real-time design

Establishing Software Standards

- Before establishing an aggressive standards development program, it is wise to formulate an overall plan that considers the available standards, the priority needs of the organization, the status of the projects, the available staff skills, and the means for standards enforcement.

Establishing Software Standards

- While it is important to establish standards, it is also important to concentrate on those standards that can be implemented in a reasonable period and that will provide the most immediate benefit to the organization.

Establishing Software Standards

- The establishment of standards starts with an examination of the organization’s standards and procedures needs.
- This should be considered in three categories, and an effort should be made to maintain a balance of emphasis among them:
  - Management and planning standards and procedures
  - Development process standards and methods
  - Tool and process standards

Establishing Software Standards

- Management and planning standards and procedures
  - Configuration management
  - Estimating and costing
  - Software Quality Assurance
  - Status reporting

Establishing Software Standards

- Development process standards and methods
  - Requirement
  - Design
  - Documentation
  - Coding
  - Integration and test
  - Reviews, walkthroughs, and inspections

Establishing Software Standards

- Tool and process standards
  - Product naming
  - Size and cost measures
  - Defect counting and recording
  - Code entry and editing tools
  - Documentation systems
  - Languages
  - Library system

Establishing Software Standards

- The standard development process
- Maintaining standards
- Enforcing standards

The Standards Development Process

- Standards development involves the following steps:
  - Establish a standards strategy that defines priorities and recognizes prior work
  - Distribute, review, and maintain this strategy
  - Select individuals or small working groups to develop the top-priority standards
  - This development effort should build on prior work where available, define the areas of applicability, specify the introduction strategy, and propose an enforcement plan.

The Standards Development Process

- The draft standards should be widely distributed and reviewed.
- The standards should be revised to incorporate the review comments and then re-reviewed if the changes are extensive.
- The standards should initially be implemented in a limited test environment.
- Based on this test experience, the standards should again be reviewed and revised.
- Implement and enforce the standards across the defined areas of applicability.
- Evaluate the effectiveness of the standards in actual practice.

Maintaining Standards

- Standards must be kept current.
- Standards should be modified and adjusted based on the experience in using and enforcing them, on the changes in available technology, and on the varying needs of the projects.
- If the standards are not maintained, they will gradually become less pertinent to working conditions and enforcement will become progressively less practical.

Maintaining Standards

- If not corrected, the standard will ultimately become a bureaucratic procedure that takes time without adding value.
- The responsibility for maintaining each standard should be assigned to an individual or group.
- This could be the SW Engineering Process Group, or some other group that has the technical capability and the management charter to handle such technical functions.

Enforcing Standards

- Standards enforcement is the basic role of the SQA organization.
- They do this with a mix of reviews and tests.
- Exhaustive reviews are most appropriate when automated tools can be used to support the monitoring process or when the standard is so critical that no single deviation is acceptable.
- Otherwise, statistical samples are sufficient unless major problems are encountered.

Enforcing Standards

- Statistical reviews are used for all other standards and procedures.
- While the level of sampling should be determined for each case, it is often possible for SQA to provide useful help to development as part of the review.
- The effective use of statistical reviews requires that SQA have control over which cases to review and management's support in follow-up actions when the review finds problems.

Standards Versus Guidelines

- A standard is appropriate when no further judgment is needed.
- Standardization makes sense when items are arbitrary and must be done uniformly or when there is one clearly best alternative.
  - The definition of coding or naming conventions
  - The selection of a programming language
  - The use of common design methods

Standards Versus Guidelines

- There are many possible selections, and there may be no clear right or wrong choice, but they must all be done the same way by everyone.
  - Appropriate subjects for standardization

Standards Versus Guidelines

- Similarly, some standards are essential to the maintenance of business or technical control.
  - Standard cost categories
  - Reporting forms
  - Change approval procedures
  - Schedule checkpoints

Standards Versus Guidelines

- The particular choices are somewhat arbitrary, but standard methods are needed because the support of several different approaches would be expensive and confusing.

Standards Versus Guidelines

- There are also many cases in which standards are totally inappropriate.
  - Typically these are cases involving technical judgment.

    The specification of limits on module size
  - Clearly it is not wise to establish a standard unless there is convincing evidence that it is always the right thing to do.

Standards Versus Guidelines

- In questionable cases, a guideline should be used or the standard should have some well-known and practical escape provisions.
- While such approaches have much the same effect as standards, they recognize that exceptions occasionally make sense.

References

Humphrey, Watts S., *Managing the Software Process*, The SEI Series in Software Engineering, Addison-Wesley, 1989. (29th Printing, May 2003) (ISBN 0-201-18095-2)