Do you really think that’s the answer to our SCM problems?

Blatant Commercialism

The Westfall Team:

- Promotes the advancement of software excellence by helping our clients achieve effective & efficient software engineering, software quality & project management practices.
- Provides training & consulting services in software engineering, software quality & project management

Open Enrollment Offering of our Software Configuration Management Course
February 3-4, 2010 in Dallas (Plano) Texas
Configuration Management Defined

Configuration management is a discipline applying technical & administrative direction & surveillance to:

- Identify the functional & physical characteristics of a configuration item
- Control changes to those characteristics
- Record & report change processing & implementation status
- Verify compliance with specified requirements
SCM Maintains a Balance

“Software configuration management is necessary to enable large teams to work together in a stable environment, yet still have the flexibility that’s needed to do creative work.”

Risk-Based Configuration Control

How much flexibility can we afford when it comes to controlling change to our software products and components?

How much stability do we need?

The answer to these, like many questions in software engineering, is that it depends on risk.
Risk Indicators

**Lower Risk**
- Small Team
- Co-Located
- Small Increment
- Frequently
- Agile
- Co-Located
- Minimal Change
- Non-Regulated
- Single Release in Production

**Higher Risk**
- Large Team
- Geographically Disperse
- Large Release
- Infrequently
- Traditional Life Cycle
- Contract Driven
- High Churn
- Extensive
- Regulated
- Multiple Releases in Production

The Good News – It’s Not All or Nothing

**Work Product**
- Creation
- Review
- Development Test
- Independent Test
- Release

**Various Levels of Control Rigor**
- Complete Flexibility
- Complete Stability
3 Techniques to Help Maintain the Balance

This presentation will discuss three techniques to help maintain the balance:

1. Selecting the appropriate type & level of control for each software artifact
2. Selecting the right acquisition point for each configuration item
3. Utilizing multiple-levels of formal control authority
What Are Configuration Items?

**Configuration item:** A work product placed under configuration management & treated as a single entity.

The following items should be placed under configuration management:

- Externally delivered software products & data
- Designated internal software work products & data
- Designated support tools used to create or support the software product
- Supplier/vendor supplied software
- Customer supplied software/equipment

---

Change Control Process

1. **Assign Author**
2. **Create/Change Configuration Item**
3. **Verification**
4. **Communication Reasons for Disapproval**
5. **Disapprove Change**
6. **CCB Decision**
7. **Wait**
8. **Defer**
9. **Issue Change Request**
10. **Release: Baseline for external use**
11. **Acquisition: Baseline for internal use**
12. **Operations**
13. **Problem/Enhancement Identified?**
   - **Yes**
     - **Problem/Enhancement Identified?**
       - **Yes**
         - **Issue Change Request**
       - **No**
         - **Release: Baseline for external use**
   - **No**
     - **Operations**
     - **Problem/Enhancement Identified?**
       - **Yes**
         - **Issue Change Request**
       - **No**
         - **Operations**
2. Configuration Item Acquisition Point

Creation or update of a Configuration Item (work product) → Quality Gate → Acquisition: Configuration Item Placed Under Configuration Control

Author(s) control changes to the Configuration Item

Formal change authority controls changes to Configuration Item
Quality Gate Examples

Examples of quality gates include the successful completion of:

- A peer review (e.g., desk check, inspection, walkthrough)
- A test activity
- A project review (e.g., phase gate review, major milestone review)
- An independent product analysis or audit

3. Multiple-Levels of Change Authority

Code example:
Membership of Multiple-Level CCB

Customers/users

- Project Manager
- Systems engineering
- Hardware development manager
- Documentation / technical publications
- Software development manager
- SQA
- SCM
- V&V

Software requirements analysts
Software architect/designer
Software engineers

Project Level CCB

Product Level CCB

Team Level CCB

Multiple-Levels of Change Authority (cont.)

Requirements example:

- Create SRS or make authorized changes
- Acquire SRS for baseline
- Release product that includes software defined by that SRS
Conclusions

Software configuration control can use multiple techniques to maintain the appropriate balance between flexibility and stability.

- The level of rigor used for each of these techniques should be determined on a project-by-project (program-by-program) basis.

- Depending on the results of a risk-based analysis of the software artifacts that will be produced by that project/program.

One of the primary roles of the software configuration management plans for a project/program is to document and communicate these decisions.

Questions?
Contact Information

Linda Westfall
3000 Custer Road
Suite 270, PMB 101
Plano, TX 75075-4499

phone: (972) 867-1172
fax: (972) 943-1484

lwestfall@westfallteam.com

www.westfallteam.com