

# A participative approach to methodology development and dissemination

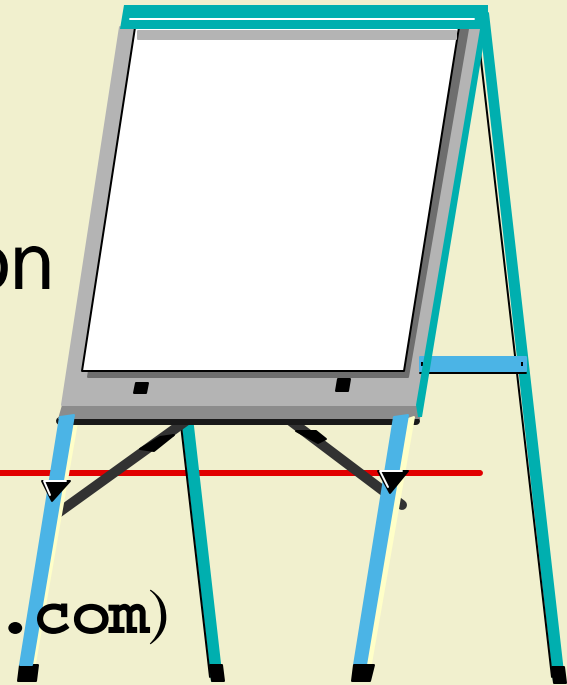
metamethodology for all  
kinds and sizes of organization

---

for C-SPIN

by Conrad Weisert ([www.idinews.com](http://www.idinews.com))

2 May, 2002



# Metamethodology issues

- What is development *methodology*?
- How is it related to:
  - ▶ standards?
  - ▶ policies?
  - ▶ good practices?
- In an organization how <sup>does</sup> <sub>should</sub> it get:
  - proposed or originated?
  - approved?
  - disseminated to the staff?
  - enforced?
  - paid for?

# Part 1:

## What is "methodology"

- Subject matter
- Scope & levels
- Two basic approaches
- Common shortcomings



# Development methodology subject matter

a broad definition

- ***Everything*** that defines how the organization develops computer application systems.

*Arguments about whether a particular collection of techniques is "a methodology" are silly!*

- Specifically . . .

# Scope: Methodology components

- The **framework**:

- ▶ Project planning & control
- ▶ System development life cycle

- Systems **analysis**

- ▶ Data definition and analysis
- ▶ Requirements specification

- **Programming** (design, coding, testing)

- ▶ Application system architecture (frameworks)
- ▶ Choice of paradigms & languages
- ▶ Program organization & structure
- ▶ Data representation
- ▶ Readability & coding conventions

*Which are independent  
of all the others?*

*& lots more -  
such as?*

# Four levels of methodology

- A mandatory **standard** (or policy):
  - Deviations must be approved in advance.
  - Mostly for choices that have an impact beyond the individual's current accountability
    - ▶ in the future
    - ▶ on other projects
- A **convention**
  - ▶ Comply unless you can show a good reason for having deviated.
- A **guideline** or recommended practice
- Helpful **information**

# Two common styles of methodology

## ■ **Military** approach (the rules)

- ▶ Emphasis on mandatory standards, enforceability
- ▶ Establishes a floor under quality
- ▶ Answers: "What must I do to satisfy Q.A.?"

## ■ **Professional** approach:

- ▶ Emphasis on flexible conventions, guidelines, and information.
- ▶ Encourages the highest quality
- ▶ Answers: "What do I need to know about \_\_\_\_\_ in order to produce high-quality results?"

*What does each assume about the audience?*

# The professional approach: assumptions about the audience

- Programmers, systems analysts, & other professionals who:
  - ▶ are keenly interested in the subject matter
  - ▶ are eager to improve their skills and learn new techniques
  - ▶ fear no threat to their creativity from sound practices
  - ▶ can grasp and apply written concepts & techniques

*Are those realistic assumptions?*

*What if they don't hold here?*



# 5 common methodology shortcomings in organizations

1. Lack of structure
2. Fragmentation
3. Structural incompleteness
4. Arbitrariness
5. Obsolescence

*How does each arise?*

*What are the consequences?*

# Common shortcomings (continued)

## 1. Lack of structure

- ▶ The information is so disorganized that staff members can't find what they're looking for
- ▶ May arise from the series of miscellaneous memos or **Technical Bulletin #55** approach

## 2. Fragmentation

- ▶ Information is scattered among multiple manuals and documents that have no clear relationship to each other (even conflict!)
- ▶ May arise from purchased methodology components or distributed support responsibility

## Common shortcomings (continued)

### 3. Structural incompleteness

- ▶ There's no obvious place to put some important piece of information
- ▶ May arise when organization's methodology structure is organized around today's tools.

### 4. Arbitrariness

- ▶ Methodology is full of *rules* and *restrictions* that have no obvious relationship to the organization's performance objectives.
- ▶ May arise from overzealous commitment to "standards".

## Common shortcomings (continued)

### 5. Obsolescence

- ▶ Most of the information was prepared years ago and no longer reflects important aspects of the environment.
- ▶ May arise when people make changes to the environment without preparing accompanying documentation.

*What about uniformity/consistency  
of style?*

## Part 2:

# How does methodology get proposed?

- Three once-popular approaches:
  - Establish representative committee(s)
  - Purchase methodology product(s)
  - Hire a world-class expert  
(the standards "czar")
- *None* of those approaches works well.

*Why not?*

# A participative approach

- *Anyone* in the organization may submit a proposed new or changed methodology component.
- *Everyone* is encouraged to do so

*In practice, only a minority will, but the whole staff will still view it as our methodology rather than something imposed from above.*

# Is it enough to invite voluntary contributions?

- Not at first.
- After establishing the infrastructure we usually identify a few highest-priority subject matter areas:
  - ▶ Solicit or assign contributions from known experts
  - ▶ These can be projects.
  - ▶

*What infrastructure?*

# Sources of new material

- **Voluntary** contributions
- **Assigned** contributions
- **Mandatory** contributions



# Part 3:

## How does proposed methodology get approved?

- Traditional approaches:
  - Representative committee(s)
  - Management review and concurrence
- Neither of those approaches works well.

*Why not?*

# Reviewers and the review process

- Designate a qualified *reviewer* in each affected organization unit.
  - ▶ M.A. distributes proposed additions/changes to the reviewers.
  - ▶ Default (no reply after a reasonable interval) is approval.
  - ▶ Reviewers rarely if ever meet as a group
- A small organization can make *everyone* a reviewer.
  - ▶ If no one objects within, say, 3 weeks the proposal becomes official methodology.

# Reviewer responses

- Accepted** (default if no response)
- Approved with reservations** as noted
- Approved subject to** specific **changes** as noted
- Disapproved** for noted reasons

*Having to state a reason makes  
many objections evaporate.*

# Objective of review process is *consensus*

- Reviewers don't *vote*; there's no tally.
- The great majority of proposals achieve consensus, either
  - ▶ initially (often by default), or
  - ▶ after one iteration of reservations or requested specific changes
- In the rare case where an impasse occurs, the M.A. can:
  - ▶ convene an ad hoc discussion among the interested, affected, and knowledgeable parties, or
  - ▶ submit the issue for management resolution

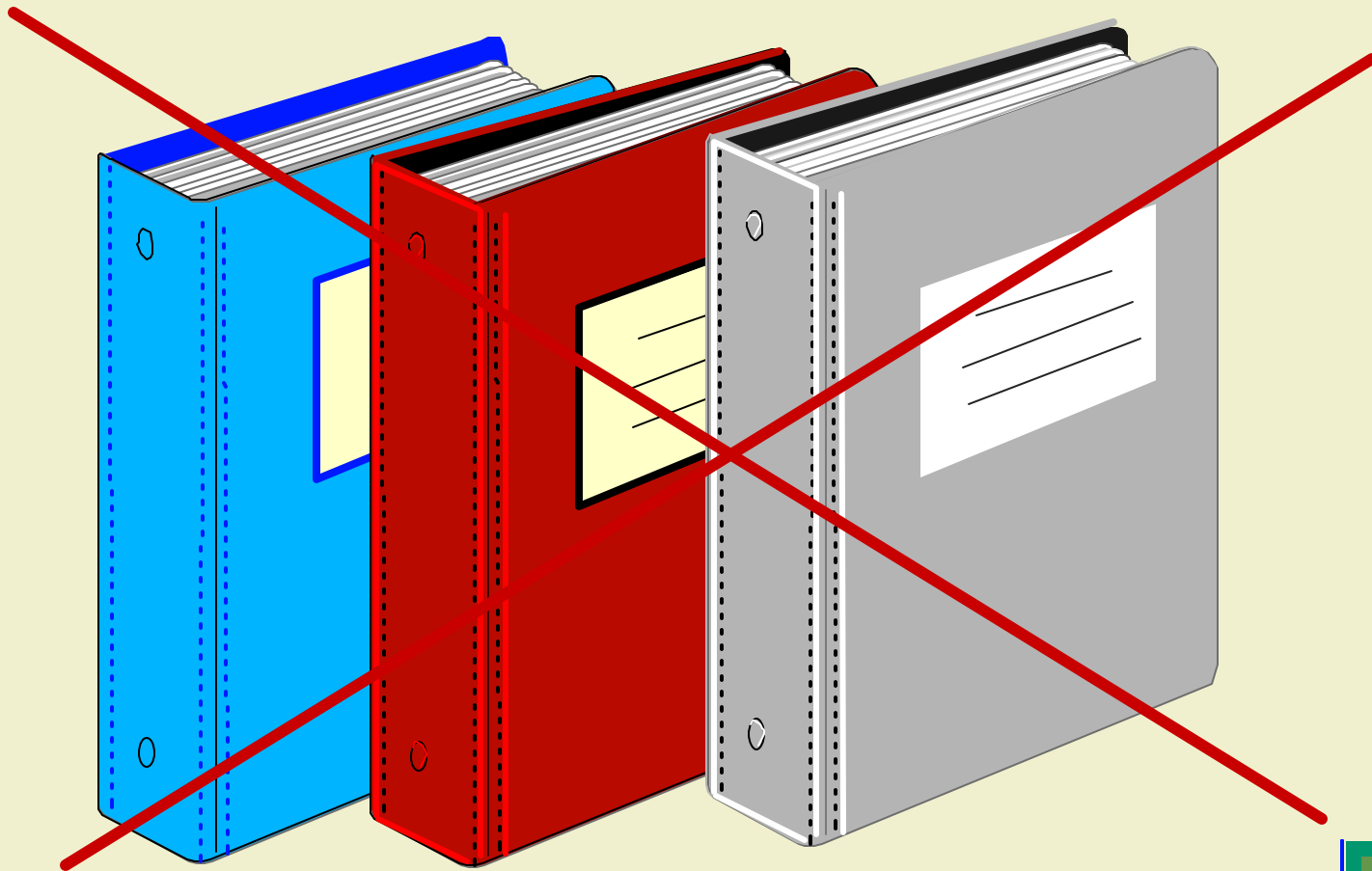
# Pitfalls of reviewing

- Perfectionism
- Tradition
- Superstarism
- Guruism

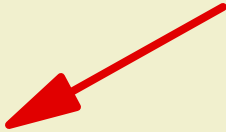
# Advice to reviewers

- Accept the proposed update if
  - a. it could be useful to some staff members *now*, and
  - b. it won't lead to future harm
- You can always proposed another improved version yourself later.
- Delay is costly

# Part 4: Disseminating and supporting the methodology



# Distribution modalities

- Written material
    - ▶ Vendor's manuals
    - ▶ Textbooks and other "literature"
    - ▶ In-house methodology documentation
  - Courses
    - ▶ In-house or public
    - ▶ Classroom or multimedia
  - Apprenticeship / mentoring
    - ▶ peer reviews, pair programming
    - ▶ internal consultants
- 



# Organizing and distributing methodology documentation

- Today's online hypertext technologies have greatly simplified both the structure and the physical distribution of methodology documentation.
  - ▶ HTTP / HTML
  - ▶ Lotus *Notes*

# Benefits of hypertext structure

- No longer a need to predetermine a hierarchical outline structure.
  - ▶ But we still need some structure *Why?*
- Updating can be nearly continuous,
  - ▶ assuring that everyone has the current version,
  - ▶ eliminating publication delays
- Easy and natural integration of:
  - ▶ subject matter
  - ▶ levels
- Many bibliographic references are now just external links.

# Distributing proposed updates for review

- Given an online distribution medium, some organizations integrate proposed updates for review into the official approved material.
  - ▶ *Proposed* (not approved) material is marked as *unofficial*, e.g. by a distinctive background page color.

# Local editions

- In a decentralized company with multiple autonomous application development organizations, some groups may want:
  - ▶ To supplement or tighten corporate standards
  - ▶ To customize practices to the local environment
  - ▶ To exempt themselves from certain elements of corporate methodology
- *How can we distribute custom versions?*
- A sensible policy:
  - ▶ ***Follow the corporate practice except where your organization has some definite need to be different.***

# Methodology and courses

- Whether public or in-house, professional development courses must be consistent with the written methodology
  - ▶ Best if the course content draws upon and strongly supports the organization's practices.
  - ▶ At the very least, no course may undermine or conflict with the written methodology.
- **M.A.** and **I.T. training director** should be organizationally close
  - ▶ Assigning both roles to a single individual has worked well

# What ever happened to quality assurance (QA)

- In many organizations during the 1990s QA evolved into little more than late-state testing & operational validation
  - ▶ QA's objective is to find "defects".
  - ▶ Deviations from the organization's standards and conventions go undetected!
- Some enlightened organizations still retain the original notion of quality
  - ▶ QA reviews should be entirely driven by the written methodology, never by personal taste.

*What other kinds of reviews do we need?*

# Part 5: Funding

## methodology infrastructure

- With a participative approach, methodology infrastructure is *very* inexpensive
  - ▶ There's no bloated bureaucracy
  - ▶ On a rational (ROI) basis, justification is virtually automatic, because of the huge multiplier.
  - ▶ Nevertheless, some highly decentralized organizations think they have no place to put it.
- Establishing methodology (standards, best practices) is not a *project*.

*Why not?*

# Essential methodology infrastructure

- Methodology administrator role
- Documentation dissemination medium
- Quality assurance mechanisms
- Reusable component library
- Professional development ("training")



# Should methodology development be a top-down or a bottom-up activity in an organization?

- *Content* (after initial "priming") is mostly bottom-up
- *Infrastructure* must be top-down

# Sustaining the infrastructure

- Many organizations establish a proactive methodology program that flourishes for a few years and then fizzles out either:
  - because budgets are leaner, or
  - because a new management team:
    - ▶ wants to take credit for eliminating "fat" and bureaucratic "red tape", or
    - ▶ wants to get rid of anything associated with the hated or discredited former regime

# Eliminating infrastructure

- Background:
  - a. Almost every organization calls for a justification (ROI) in order to *establish* new supporting infrastructure.
- Weisert's rule:
  - a. It's just as important to justify *dismantling* infrastructure. Upper management should expect and demand impact assessment.
  - b. Don't change anything before you understand it.

*Remember the Y2K crisis?*

# Advice for lean times

- Under no circumstances drop the methodology administrator role or methodology infrastructure funding from the budget, even for just one year.
- Orientation of new employees must always strongly endorse the established methodology framework.

*This advice is easier to  
follow today than in 1985.*