Software Best Practices Clearinghouse

Promoting Adoption and Effective Implementation

Kathleen Dangle
Fraunhofer Center for Experimental Software Engineering

Thomas McGibbon
ITT Industries/Data & Analysis Center for Software (DACS)

Richard Turner
The George Washington University
Presentation Objectives

• **Share** with you our thinking on why we believe programs face challenges implementing best practices and how we overcome those challenges

• **Inform** you about the Best Practices Clearinghouse Initiative

• **Encourage** you to think about your experiences with considering or implementing best practices

• **Request** your feedback and motivate you to get involved
How Do We Encourage Broader Use of Best Practices?

Through the Best Practices Clearinghouse

- Promote and assist in the adoption and effective utilization of “best practices”
- Provide central access to validated, actionable practice information
- Target the needs of the Department of Defense software acquisition and development community
Implementation Barriers

- Programs are aware of “best practices,” but they don’t often choose to implement them
  - Too many lists to choose from
  - No basis for selecting specific practices
  - Proof of effectiveness is not generally available
  - Not easy to see connection between practices and specific program risks or issues
  - Practice’s success factors not well understood
  - Resources are limited and the return on practice investment is unknown
  - Implementation guidance is inadequate
Traditional Best Practices

- Are disciplines rather than specific practices (e.g., Risk Management)
- Have problematic descriptions
  - If descriptions too generic or abstract, hard to apply; if too context specific, don’t seem relevant
  - Implementation directions insufficient, ineffective, imprecise
  - Rarely supported by data
- Take energy and resources to implement, but benefits may come (much) later or are hard to quantify
- Implementation does not always work
  - Often depend on other practices
  - Are not implemented as designed
  - Depend on project context (size, complexity, life-cycle phase)
What Do We Mean By ‘Supported By Data’?

- Example: NASA Software Engineering Laboratory Ground Support Systems Software Development
  - Used experiments and data to evaluate, select, implement and track the impact of development practices
  - By feeding back actual performance data into their work, and using only practices their data showed effective, they:

  Decreased Development Defect rates by

  Reduced Cost by

  Improved Reuse by

  Increased Functionality five-fold (1976 - 1992)
Practice Analysis Examples

- **Best practice: Smaller modules have less defects**
  - Reality: Observation and analysis showed sweet spot

![Graph showing fault rate vs size/complexity]

- **Best Practice: Early detection of defects**
  - Initial experience: late detection >100X more expensive
  - New data showed
    - 100X still valid for severe defects
    - However, only 2X more expensive for less severe defects
    - Business model drives acceptance of late costs
The Clearinghouse Vision

- *The best practice resource for the Department of Defense*
- Based on empirical evidence
- Validated practice information provides level of confidence
- Leverages existing best practices and centralizes access to them
- Captures cost, benefits, context, latency
- Supports user-driven selection of relevant practices
- Provides step-wise implementation guidance and expert assistance
- Tracks and measures results
Key Strategies to Overcome Challenges

- **User-focused** access and information infrastructure
- **Empirically based** Information in the repository
- The building block of each practice or set of practices is a “story”
- A set of stories are synthesized into a profile
- Details of the practice are provided on demand
- A type of color code scheme provides a quick and easy way of understanding the level at which the practice is well-proven or robust
Delivery Infrastructure Focused on Users

• Easy to use, informative tools for best practices selection and implementation support
  - Practices suggested by goal, risk, phase, program size
  - Implementation ordering for multiple practices
  - Evolution from basic through advanced practices
  - Flexible search mechanisms

• Active community involvement and links to expertise
  - Acquisition Community Connection (nee PM CoP)

• Dissemination of Clearinghouse latest information through widely-used venues: courses, workshops, articles, conference tutorials
Exploiting Sources of Information

- **Identify and utilize what we already know**
  - Mine best practices and lessons learned repositories (from the Services, Agencies, FFRDCs, DAU, Academic Institutions, DACS Gold Practices, Industry, literature, etc.)
  - Cultivate relationships with practice experts and researchers
  - Gather experiences on specific programs

- **Make it readily accessible**
  - One central entry point to organized information
  - Not re-publish what is already there, but provide links

- **Make it easy to use**
  - Extract key information from more detailed sources
  - Provide visual cues and progressively more detailed information

- **Keep it current**
  - E-workshops support practice identification and validation
  - User feedback
  - Ongoing study, conferences, workshops, symposia
## Best Practices Vetting Process

Each cycle allows more experience to be gathered and processed, leading to better characterization of the practice, improved recommendations, and more dependable implementation guidance.

### Practice/packaging maturation cycle

<table>
<thead>
<tr>
<th>Identification</th>
<th>Characterization</th>
<th>Analysis &amp; Synthesis</th>
<th>Validation</th>
<th>Packaging &amp; Dissemination</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inputs:</strong></td>
<td><strong>Inputs:</strong></td>
<td><strong>Inputs:</strong></td>
<td><strong>Inputs:</strong></td>
<td><strong>Inputs:</strong></td>
</tr>
<tr>
<td>Leads to practices</td>
<td>Set of candidate practices and rationale for consideration</td>
<td>Detailed set of candidate practices</td>
<td>Sets of practice data; validation criteria</td>
<td>Sets of practice data; validation criteria</td>
</tr>
<tr>
<td><strong>Activities:</strong></td>
<td><strong>Activities:</strong></td>
<td><strong>Activities:</strong></td>
<td><strong>Activities:</strong></td>
<td><strong>Activities:</strong></td>
</tr>
<tr>
<td>• Collect</td>
<td>• Aggregate stories, create profile of practice</td>
<td>• Check outputs from previous phases</td>
<td>• Packaging</td>
<td>• Packaging</td>
</tr>
<tr>
<td>• Categorize</td>
<td>• Populate the repository</td>
<td>• Publishing</td>
<td>• Publishing</td>
<td>• Publishing</td>
</tr>
<tr>
<td>• Filter</td>
<td>• Identify/define Interrelationships</td>
<td>• Promoting</td>
<td>• Promoting</td>
<td>• Promoting</td>
</tr>
<tr>
<td>• Synthesize</td>
<td><strong>Outputs:</strong></td>
<td>• Provide user help</td>
<td>• Providing user help</td>
<td>• Providing user help</td>
</tr>
<tr>
<td>• Prioritize</td>
<td>Candidate set of practices</td>
<td>More detailed set of candidate practices with “stories”</td>
<td>Approve practices via panel of experts</td>
<td>Repository update</td>
</tr>
<tr>
<td><strong>Outputs:</strong></td>
<td><strong>Outputs:</strong></td>
<td><strong>Outputs:</strong></td>
<td><strong>Outputs:</strong></td>
<td><strong>Outputs:</strong></td>
</tr>
<tr>
<td>Leads to practices</td>
<td>Candidate set of practices</td>
<td>More detailed set of candidate practices with “stories”</td>
<td>Validated practices</td>
<td>More detailed set of candidate practices with “stories”</td>
</tr>
</tbody>
</table>

### Possible practice validation coding

- **Proven**
- **Consistent results**
- **Initial validation**
- **Nominated**
**Case Study # 24**

**Best practice**

Formal inspections


**Source**

The use of software inspections will ensure a high level of system quality.

**Lesson Learned**

Attention must be paid that inspections are practiced correctly, with appropriate formality, to ensure defect removal benefits.

**Breakdown in the use of inspections**

- Contrary to typical practice, there was not a requirement for a navigation (end-user) representative to be present at any of the walkthroughs or the acceptance test.
- The Sm forces software not mission critical, which reduced the number of reviews done on the software compared to mission critical.

### BP Interrelationships

**Architecture-First Approach**

- **Ensure Interoperability**
- **Develop/Maintain A Life Cycle**
- **Business Case**
- **Common Management**
- **And Manufacturing Systems**
- **Commercial Specifications**
- **And Standards/Open Systems**
- **Capture Artifacts In Rigorous Model-Based Notation**
- **Assess Reuse Risks and Costs Agreement On Interfaces**
- **Acquisition Process Improvement**
- **Requirements Trade-Offs Negotiations**
- **Plan for Technology Insertion**
- **Manage Requirements**
- **Leverage COTS/NDI**
- **Integrated Product And Process Development (IPPD)**
- **Independent Expert Reviews/SCEs**
- **Formal Risk Management**
- **Enables Provide a basis for decisions Documents/communicates the architecture Requires architecture be evaluated by Assess the value of adopting Is a required part of Business goals  & requirements drive architecture decisions Risks are identified and drive decisions Is necessary for**

### Experience data

**Conceptual BP Information**

**Characteristic data**

<table>
<thead>
<tr>
<th>No.</th>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Indicators</td>
<td>Types that support implementation of this practice would be advantageous.</td>
</tr>
<tr>
<td>2</td>
<td>Communications</td>
<td>Types that support against implementing this practice.</td>
</tr>
<tr>
<td>3</td>
<td>Appropriate Candidates</td>
<td>Types of programs that would benefit from implementation.</td>
</tr>
<tr>
<td>4</td>
<td>Inappropriate Candidates</td>
<td>Types of programs where this practice would be useless or harmful.</td>
</tr>
<tr>
<td>5</td>
<td>Enablers</td>
<td>List barriers to implementation under each category (categories described in instructions).</td>
</tr>
<tr>
<td>6</td>
<td>Enablers</td>
<td>List enablers which could support implementation under each category (categories described in instructions).</td>
</tr>
</tbody>
</table>

**Implementation data/ guidance**

**Inspection process overview**

**Phase 1: Planning**

Inspectors should have vested interests in work product

Inspectors should invest no more than 15% of their time in inspections (don't overwork good inspectors!)

**Phase 2: Preparation**

Inspectors should spend at least as much time in preparing as is required for the inspection meeting.

Provide adequate lead time for inspectors to perform preparation (3 - 5 work days)
Example Tool for Practice Selection & Investigation

Life Cycle Phase: CTD
Risks/Issues: Limited SW productivity
Validation Coding: Proven
Mitigation: Architect SW for parallel development
DACS Gold Practices

- Initiative began in early-2002, extending previous best practice research
- Objectives:
  - Disseminate consistent, easy-to-understand, value-added best practice information
  - Gather user experience on best practice information
- 35 practices identified; 4 currently described
- Relationship to Clearinghouse
  - Initial information source for Clearinghouse
  - Clearinghouse activities will inform and improve Gold Practice products
How Can You Get Involved?

• *Let us know your needs by*
  
  - Identifying your best practices *lists* and *sources* of guidance for their use
  - Sharing your *experiences* & lessons learned in implementing best practices
  - Volunteering to help us define the services and capabilities of the Clearinghouse
  - Participating in surveys, e-workshops and other events - See [http://iac.dtic.mil/dacs](http://iac.dtic.mil/dacs) for more information

• *Participate in the next session, “Software Acquisition Best Practices Workshop”*
The Best Practices Clearinghouse – In Summary

- Centralized resource
- Lessons learned in application of practices
- Empirically based, Experiences provided
- Acquisition and development practices
- Repository of vetted practices
- Important insight
- Not just another list; Not just a website
- Guidance on selection
- Help provided through multiple services
- Outreach to user community
- Useful information
- Search capabilities
- Easy to use & informative tools
Contact Information

Kathleen Dangle
Fraunhofer Center for Experimental Software Engineering
kdangle@fc-md.umd.edu
301-403-8973

Thomas McGibbon
ITT Industries/ Data & Analysis Center for Software (DACS)
Tom.McGibbon@itt.com
315-334-4933

Richard Turner
The George Washington University
Rich.Turner.CTR@osd.mil
703-602-0851 x124