Making Process Improvement Work

A Concise Action Guide for Software Managers and Practitioners

Neil Potter
Mary Sakry

The Process Group
help@processgroup.com
www.processgroup.com
Session Agenda

1. Introduction
2. Scoping Improvement
   • Using goals and problems
3. Developing an Action Plan
   • Using the CMMI framework
The “Classic” Approach to PI

Process-centric improvement
- SEI CMMI
- ISO9001
- Bellcore

It can work!
- High risk of failure
Starting point

Common result: Lost in the trees
A Solution

Goal-problem-centric improvement

Goals and problems can be used to scope and sequence the improvement effort
Goal

Problem areas

Starting point

- Goal actions
- Improvement actions
Frameworks

- Frameworks provide an optional source of improvement ideas, e.g.,
  - Life cycle
  - SEI CMMI
  - ISO9001
  - Bellcore

- In this workshop, either use:
  - No framework
  - Current organization’s life cycle and defined practices
  - Published framework
Developing a Plan

• **Scope the Improvement**
  1. Establish plan ownership
  2. State the major goals and problems
  3. Group the problems related to each goal
  4. Ensure that the goals and problems are crystal clear and compelling
  5. Set goal priorities
  6. Derive metrics for the goals

• Develop an Action Plan

• Determine Risks and Plan to Mitigate
1. Establish Plan Ownership

• The plan meets the owner’s needs, e.g.,
  – Business goals and problems
• The owner can be a project manager, program manager, senior manager, or division head
• The primary owner ≠ EPG or QA group
  – Support functions can share ownership
• Different individuals can be responsible for each section of the plan

EPG = engineering process group
QA = quality assurance group
2. State the Major Goals and Problems

Example Goals

1. Create predictable schedules
2. Successfully deliver product X
3. Reduce rework
4. Improve the performance of our core product
5. Keep customers happy
6. Keep making a profit
State the Major Goals and Problems

Example Problems

1. Need better requirements. Requirements tracking not in place. Changes to requirements are not tracked; code does not match specification at test time.
3. Quality department does not have training in product and test skills.
4. Unclear status of changes.
5. Lack of resources and skills allocated to design.
10. Wrong files (for example, dynamic link libraries) are put on CD. Unsure of the correct ones.
11. Revising the project plan is difficult. Items drop off, new things are added, plan is out of date.
12. We don’t understand our capacity and do not have one list of all the work we have to do.
13. Schedule tracking and communication of changes to affected groups is poor.
3. Group the Problems Related to Each Goal

- Simplify the list by grouping the problems that prevent each goal from being achieved.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Problem</th>
<th>Problem Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Create predictable schedules</td>
<td>Problem 11</td>
<td>Revising the project plan is difficult. Items drop off, new things are added, plan is out of date.</td>
</tr>
<tr>
<td></td>
<td>Problem 12</td>
<td>We don’t understand our capacity and do not have one list of all the work we have to do.</td>
</tr>
<tr>
<td></td>
<td>Problem 13</td>
<td>Schedule tracking and communication of changes to affected groups is poor.</td>
</tr>
</tbody>
</table>
Group the Problems Related to Each Goal - 3

<table>
<thead>
<tr>
<th>Goal</th>
<th>Problem</th>
<th>Problem Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. Reduce rework</td>
<td>Problem 3</td>
<td>Quality department does not have training in product and test skills.</td>
</tr>
<tr>
<td></td>
<td>Problem 4</td>
<td>Unclear status of changes.</td>
</tr>
<tr>
<td></td>
<td>Problem 5</td>
<td>Lack of resources and skills allocated to design.</td>
</tr>
<tr>
<td></td>
<td>Problem 9</td>
<td>Defect repairs break essential product features.</td>
</tr>
<tr>
<td></td>
<td>Problem 10</td>
<td>Wrong files (for example, dynamic link libraries) are put on CD. Unsure of the correct ones.</td>
</tr>
</tbody>
</table>
4. Ensure That the Goals and Problems Are Compelling

If there is a big enough reason, people will change in a heart beat!

- Crisis
- Business loss
- Unhappy customers

- High development cost
- Low profit
- Leadership

- New business
- Increase volume
- Reduce costs
Ensure That the Goals and Problems Are Compelling - 2

• Example goals that are not compelling:
  – Document all processes.
  – Develop a detailed life cycle.
  – Establish a metrics program.

• Example goals that are more compelling:
  – Deliver product X by Dec 15th.
  – Increase product quality to a maximum of 10 defects per release, gaining back customers X, Y, and Z, and increasing our market share by 10 percent.
  – Reduce rework to 5 percent of project effort. Use that time to create new product Y.
  – Improve schedule prediction to \( \pm 5 \)-day accuracy, eliminating forced cancellation of vacations.
Develop an Action Plan

• **Develop an Action Plan**

1. Enumerate actions using brainstorming and a process framework
   » 1a. What actions are needed to address the problems and achieve the goals?
   » 1b. If a process improvement framework is being used, which elements will help the problems and goals listed?

2. Organize the action plan based on the goals and problems

3. Add placeholders for checking progress and taking corrective action
# 1a. Actions for Two of the Problems

<table>
<thead>
<tr>
<th>Problem</th>
<th>What actions are needed to address the problems and achieve the goals?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Changing requirements</td>
<td>Baseline the requirements before design commences</td>
</tr>
<tr>
<td></td>
<td>Only allow changes to the application interface, not to the kernel routines</td>
</tr>
<tr>
<td></td>
<td>Improve the library control system to minimize version control errors</td>
</tr>
<tr>
<td></td>
<td>Investigate requirements management tools</td>
</tr>
</tbody>
</table>
1b. Framework Elements for Two of the Problems

<table>
<thead>
<tr>
<th>Problem</th>
<th>Which elements will help the problems and goals listed?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Changing requirements</td>
<td>Develop an understanding with the requirements providers on the meaning of the requirements. (REQM sp1.1)</td>
</tr>
<tr>
<td></td>
<td>Assign responsibility and authority for performing the REQM process. (REQM gp2.4)</td>
</tr>
<tr>
<td></td>
<td>Track change requests for the configuration items. (CM sp2.1)</td>
</tr>
</tbody>
</table>

REQM = Requirements Management. CM = Configuration Management
Progress on Chosen Framework

Example Goals
1. Create predictable schedules
2. Successfully deliver product X
3. Reduce rework
4. Improve the performance of our core product
5. Keep customers happy
6. Keep making a profit

Example Problems
1. Need better requirements. Requirements tracking not in place. Changes to requirements are not tracked; code does not match specification at test time.
3. Quality department does not have training in product and test skills.
4. Unclear status of changes.
5. Lack of resources and skills allocated to design.
10. Wrong files (for example, dynamic link libraries) are put on CD. Unsure of the correct ones.
11. Revising the project plan is difficult. Items drop off, new things are added, plan is out of date.
12. We don’t understand our capacity and do not have one list of all the work we have to do.
13. Schedule tracking and communication of changes to affected groups is poor.
### Example Improvement Plan

<table>
<thead>
<tr>
<th>Primary Goal and Intermediate Goals</th>
<th>Purpose of Goal (Why do you want to achieve the goal?)</th>
<th>Actions</th>
<th>Priority (*=essential)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce product development cycle to six to nine months for product X.</td>
<td>Deliver earlier than competition.</td>
<td>Only allow changes to the application interface, not the kernel routines.</td>
<td>1*</td>
</tr>
<tr>
<td>Manage changing requirements (based on problem 1).</td>
<td>Prevent schedule slips resulting from expensive scope changes.</td>
<td>Assign responsibility and authority for performing the REQM process.</td>
<td>2*</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Check progress and take corrective action.</strong></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve the library control system to minimize version control errors.</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Investigate requirements management tools.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Track change requests for the configuration items.</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Develop an understanding with the requirements providers on the meaning of the requirements.</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Baseline the requirements before design commences.</td>
<td>6</td>
</tr>
</tbody>
</table>

**Step 3:** Add placeholder for checking progress and taking corrective action.


References

22. ROI information: http://www.processgroup.com/resources.htm (see ROI Data)