Tool and Data Interoperability in the SSE System

Chuck Shotton
PRC
11/10/88
Overview

- Industry Problems with Program and Data Interoperability
- SSE System Interoperability Issues
- SSE Solutions to Tool and Data Interoperability
- Attaining Heterogeneous Tool/Data Interoperability
Software Development Methods

- Representations
- Deriving the representations
- Examining the representations
Goals

- Maintain separation of methods from tools supporting the methods
- Point of view of methods and tool users, not tool-builders
- Separate classification from evaluation
- Repository for information
- Determine "gaps" in methods and tools
SPIRAL MODEL OF SOFTWARE PROCESS
<table>
<thead>
<tr>
<th>Level</th>
<th>Characteristic</th>
<th>Key Problem Areas</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimizing</td>
<td>Improvement fed back into process</td>
<td>Automation</td>
<td>Productivity &amp; Quality</td>
</tr>
<tr>
<td>Managed</td>
<td>(quantitative) Measured process</td>
<td>Changing technology</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Problem analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Problem prevention</td>
<td></td>
</tr>
<tr>
<td>Defined</td>
<td>(qualitative) Process independent of individuals</td>
<td>Process measurement</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Process analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quantitative quality plans</td>
<td></td>
</tr>
<tr>
<td>Repeetable</td>
<td>(intuitive) Process dependent on individuals</td>
<td>Training</td>
<td>Risk</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Technical practices</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• reviews, testing</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Process focus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• standards, process groups</td>
<td></td>
</tr>
<tr>
<td>Initial</td>
<td>(ad hoc / chaotic)</td>
<td>Project management</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project planning</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configuration management</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software quality assurance</td>
<td></td>
</tr>
</tbody>
</table>
Process Definition

- A sequence of life cycle tasks, which when properly executed produces the desired result

- An effective process must consider
  - the relationships of all the required tasks
  - the tools and methods used
  - the skills, training, motivation, and management of the people involved
Waterfall

CONCEPT

FEASIBILITY STUDY

INTERNAL SYSTEM REQUIREMENTS REVIEW

SYSTEM REQUIREMENTS REVIEW

SYSTEM DESIGN REVIEW

PRELIMINARY DESIGN REVIEW

DESIGN

CODING AND CHECKOUT

TESTING

INTEGRATION

OPERATIONAL TEST AND EVALUATION

SYSTEM PERFORMANCE TEST

FUNCTIONAL CONFIGURATION AUDIT

DEPLOYMENT MAINTENANCE

SYSTEM PERFORMANCE TEST

FORMAL QUALIFICATION REVIEW

PHYSICAL CONFIGURATION AUDIT

MILESTONE I PROGRAM DECISION

MILESTONE II RATIFICATION DECISION

MILESTONE III PRODUCTION DECISION

MILESTONE 0 PROGRAM INITIATION
Strategy

Promote the evolution of software engineering from an ad hoc, labor-intensive activity to a managed, technology-supported discipline.
Implementation of Strategy

- Put process under management control
  - define
  - measure
  - optimize

- Adopt appropriate methods

- Insert technology that provides automated support for the process and methods

- Collect automated tools into an integrated environment

- Educate people
CASE

Components

- Process
- Methods
- Computers
- Tools
- Support environments
- Engineers

Currently the engineers are the essential integrating factors tying all these components together.

The engineers today empower the tools versus the tools empowering the engineers.