
Team Software Process

Integrantes:

Ángel Asís Reyes

Luisa Mariana Pulido González

Miriam Lilly Martínez Guzmán

Víctor Manuel Arredondo Reyes


¿Qué es TSP?


- El **Team Software Process** es un proceso que provee mejoras en el proceso de software, brindando un contexto disciplinado para el trabajo de ingeniería. Este tiene como base el **PSP**(Personal Software Process).

¿Por qué fue creado?


- Fue diseñado para que tanto proyectos pequeños como grandes puedan ser correctamente administrados y consigan un producto resultante de calidad. Su objetivo es construir y guiar a los equipos de proceso de software.

¿Por qué es necesario un proceso?

- 
- El desarrollo de sistemas es una actividad en equipo y la efectividad del mismo determina en gran manera la calidad de la ingeniería.

- 
- Debido a que el equipo se enfoca mucho mas en el producto y deja de lado el proceso.

- 
- Hasta los miembros del equipo se pueden volver obstáculos si no hay un proceso.

- 
- Los equipos gastan mucho tiempo en lograr un manejo de el mismo y surgen preguntas:
 - ¿Cómo repartir el trabajo?
 - ¿Cómo coordinar tareas?
 - ¿Cómo asignarse responsabilidades?
 - ¿Cómo medir avance?

Condiciones Para Trabajar En Equipo

Un equipo tiene:

- Se compone por al menos dos personas
- Miembros que trabajan en pos de una meta en común
- Roles específicos asignados para cada persona

Completar la misión requiere de algún modo, la dependencia entre los miembros del equipo

- **¿Será necesario asignar roles?**
- Si, debido a que:

Generan un sentido de pertenencia

Guiar al equipo en el cómo hacer su trabajo

Prevén:

- Conflictos
- Duplicación de trabajo

Proveen un grado de control sobre su trabajo

Equipos Eficientes

- Los equipos que son realmente eficientes tienen ciertas características en común:

Sus miembros poseen habilidades

Sus metas del equipo son importante, definida, visible y realista

Sus recursos de equipo son los adecuados

Sus miembros están motivados y se comprometen a trabajar en la meta del equipo

Existe cooperación y apoyo de parte de todos

Sus miembros manifiestan disciplina de trabajo

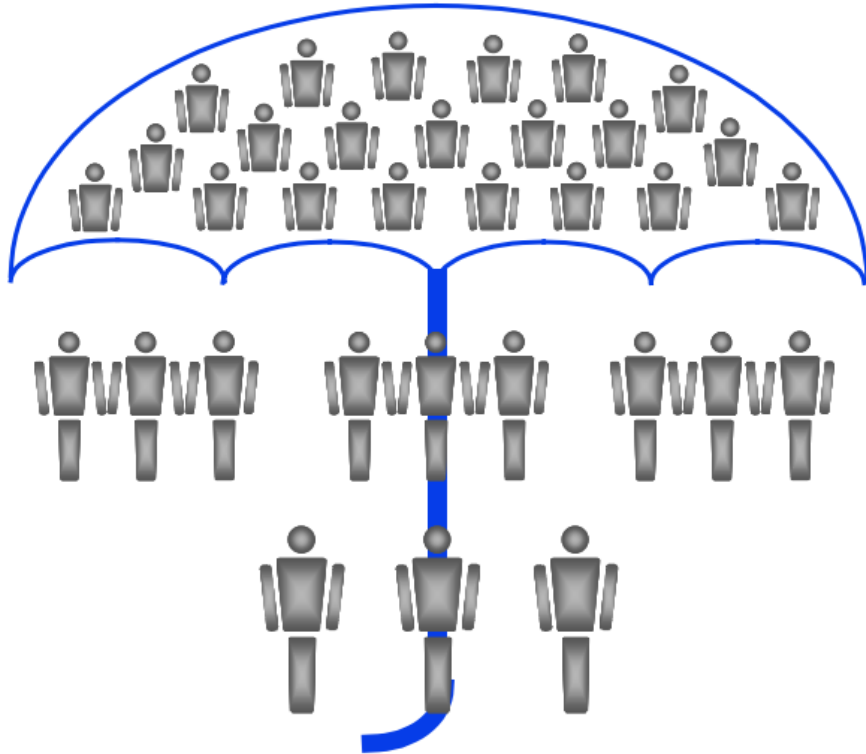
Poseen habilidad de inovacion

Construyendo Un Equipo Efectivo

- Los miembros del equipo establecen metas en común y sus respectivos roles
- El equipo desarrolla una estrategia previamente aceptada por todos
- Los miembros definen un proceso en común para su trabajo
- Todos los miembros participan en producir el plan, y cada miembro conoce su rol en él
- El equipo negocia el plan con la administración
- La administración revisa y acepta el plan negociado
- Los miembros realizan el trabajo de acuerdo al plan, comunicándose seguido y libremente
- El equipo forma un grupo cohesivo: los miembros cooperan, y todos están comprometidos a alcanzar la meta
- Los ingenieros conocen su status, son retroalimentados, tienen liderazgo que los mantiene motivados

¿Dónde Entra El Team Software Process?

- Un proceso operacional es como un guión diseñado para ser usado por el equipo mientras trabaja.
- El TSP provee un proceso operacional para guiar a ingenieros y encargados a través la construcción del equipo, establece un ambiente de trabajo efectivo y guía a equipos usando un proceso definido y un plan para seguir ese proceso, mejorando enormemente la eficiencia.
- Team Software Process une los principios del Personal Software Process (PSP) y el Capability Maturity Model (CMM). En esencia, CMM y PSP proveen el contexto y las habilidades para una ingeniería efectiva mientras el TSP guía a los ingenieros mientras **realizan** dicho trabajo.



CMM - Improves organization's capability; management focus.

TSP - Improves team performance; team and product focus.

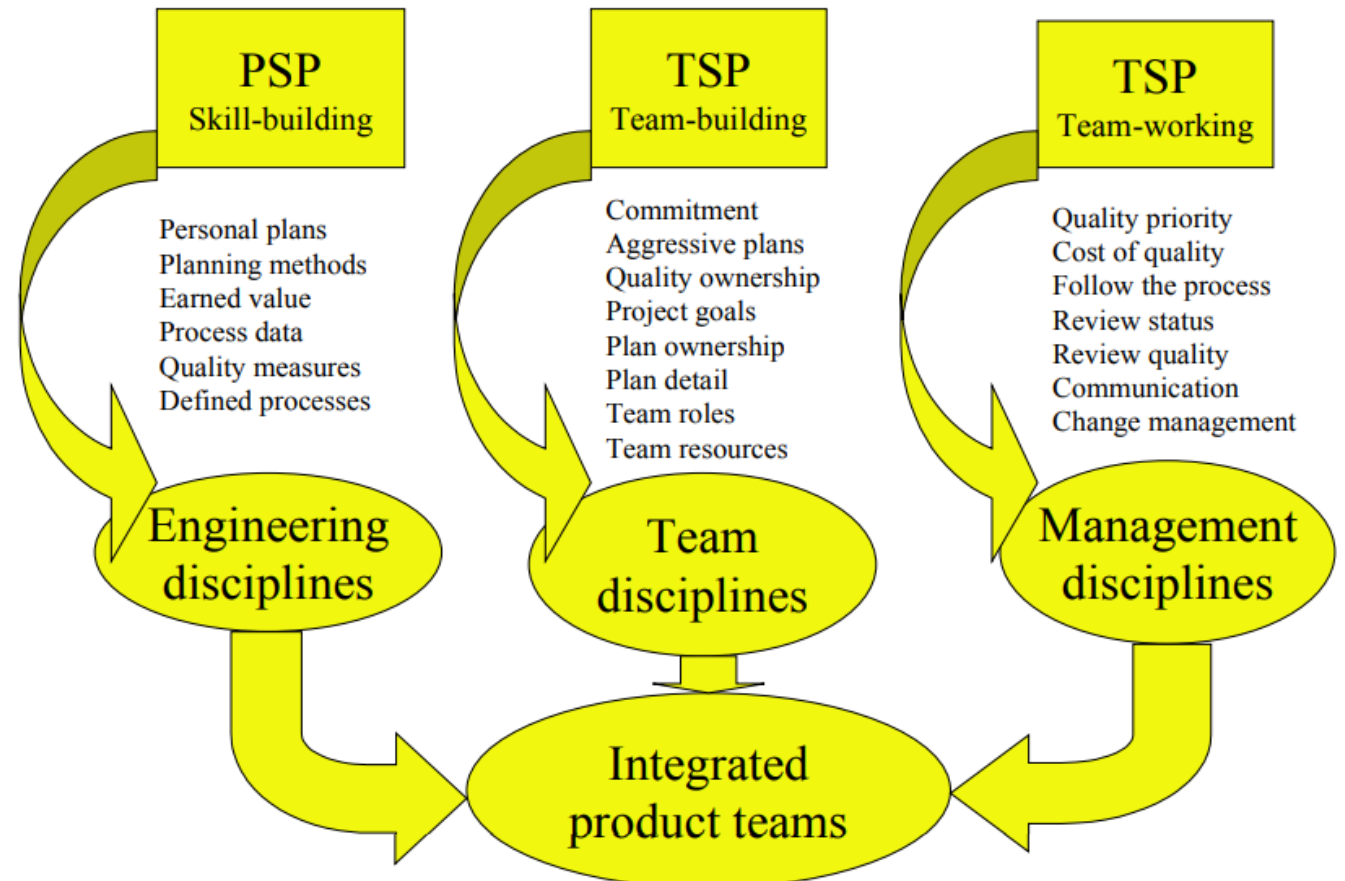
PSP - Improves individual skills and discipline; personal focus.

Métodos De Mejora De Procesos

Diseño de TSP

Requiere:

- Ingenieros disciplinados, entrenados en PSP.
- Conocimiento previo en diseño de software y manejo de requerimientos.
- Conocimiento en administración de la configuración, manejo de proyectos y pruebas de software.



¿Cómo construir equipos de TSP?

- Se deben seguir ciertas medidas divididas por:

Metas



- ☐ Definir y aceptar metas similares
- ☐ Definición participativa

Roles



- ☐ Definir responsabilidades entre todos los miembros de equipo

Planes



- ☐ Agregar estrategias para lograr objetivos
- ☐ Dividir en partes el trabajo total

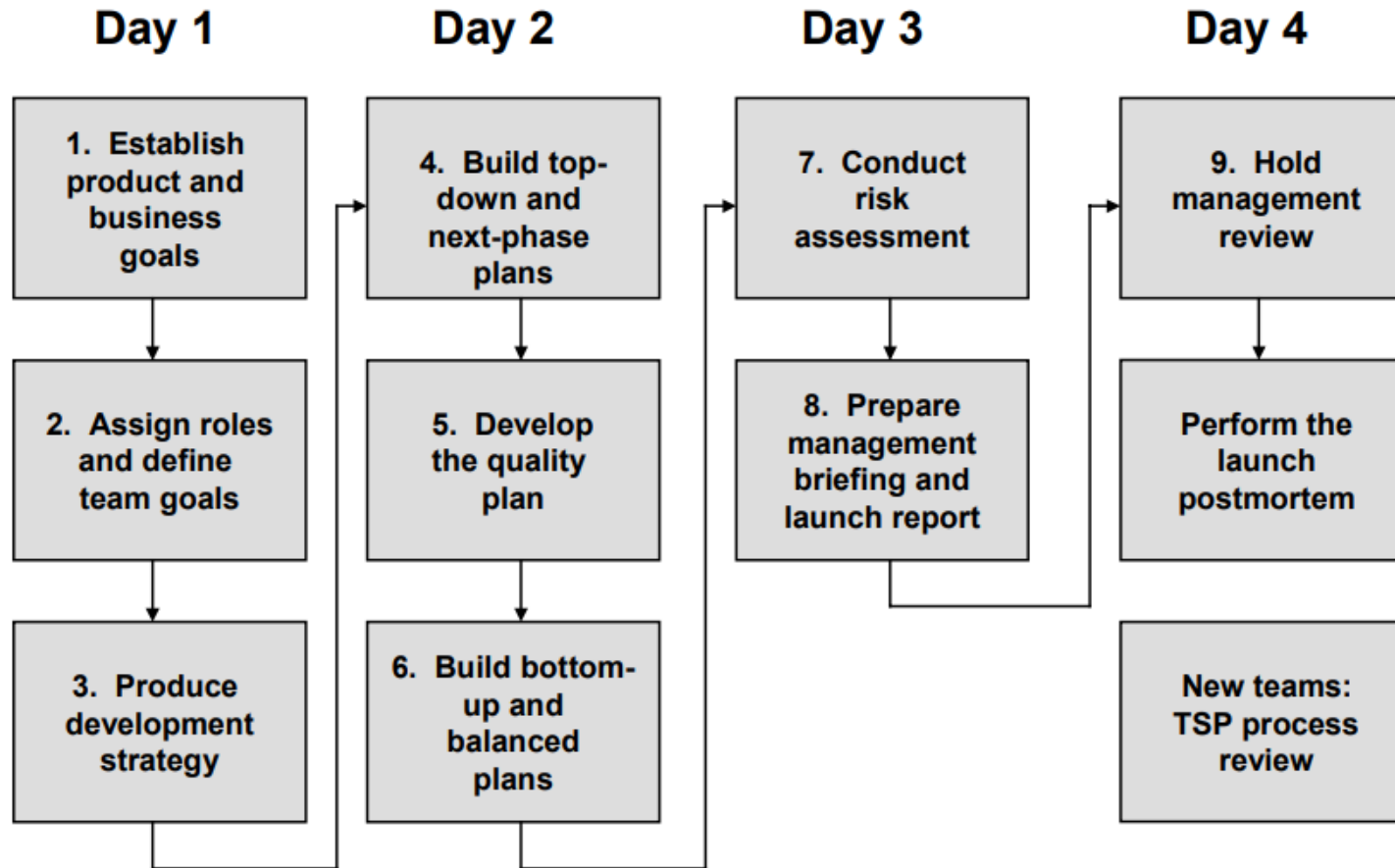
Comunicación




- ☐ Conocer el estado actual de trabajo de cada uno y constantemente
- ☐ Mantener seguimiento de acuerdo a los planes

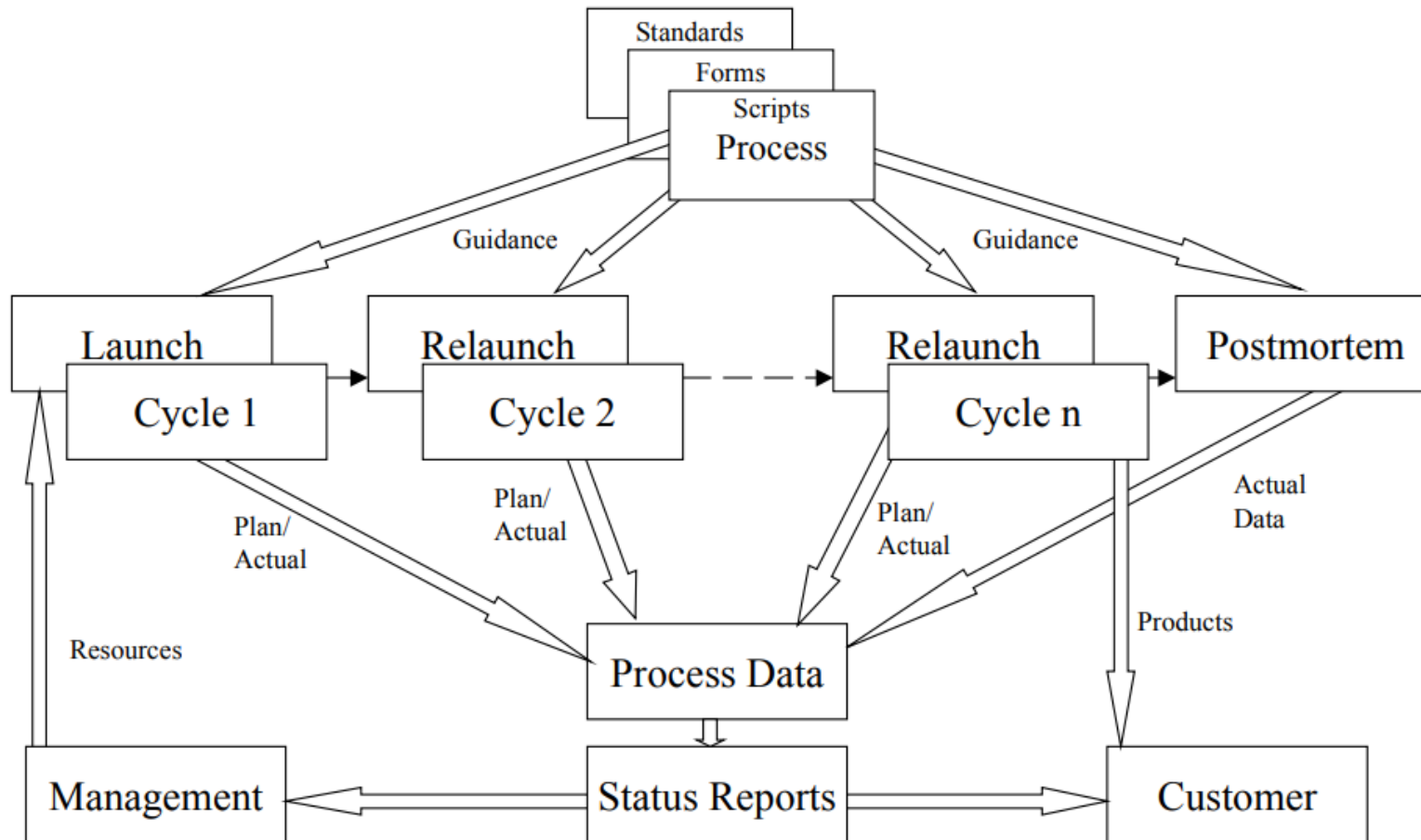
Estructurando TSP

- En el TSP, el proceso de construcción del equipo es de cuatro días y se le llama **Team Launch**. En él, todos los miembros del equipo desarrollan la **estrategia**, formulando un **plan detallado** y un **plan general** para los próximos 3 o 4 meses, así como el **proceso y plan** para hacer su proyecto. Al completar el lanzamiento (launch), el equipo sigue el proceso definido para el trabajo.



TSP Launch Process

- 
- El team launch es realizado periódicamente porque el TSP es un proceso **iterativo**. De este modo cada fase o ciclo puede ser planeado basado en el conocimiento obtenido en el ciclo anterior. Igualmente, el proceso de lanzamiento es requerido para actualizar los planes detallados de los ingenieros, ya que estos son precisos solo por unos meses.



Ciclo De Desarrollo de TSP

Guiando Al Equipo

- El líder del equipo es responsable de guiar y motivar a los miembros, manejar asuntos con los clientes y lidiar con el management.
- Por sobre todo, el líder de equipo debe mantener la motivación del mismo y la energía, para asegurarse de que el trabajo se realiza tan efectivamente posible.



¿Qué incluye ser el líder?

- Dirigir el trabajo día a día
- Proteger los recursos del equipo
- Resolver problemáticas del equipo
- Conducir reuniones con el equipo
- Reportar el trabajo
- Mantener la disciplina

Manejo De Calidad



¿Por qué se busca manejarla?

- Todos quieren manejar la calidad pero la realidad es que la mayoría de equipos no saben como como administrar la calidad de sus productos.
- Lo cierto es que no existen métodos generales que eviten totalmente los defectos porque las personas desarrollan software y cometen errores



Manejo De Calidad

¿Cómo manejarla?

- Para hacerlo, los equipos deben establecer métricas de calidad, fijarse metas, establecer los planes para cumplir dichas metas, medir el progreso basándose en los planes y tomar acciones que contrarresten los efectos negativos cuando las metas no sean alcanzadas
- Una vez se ha generado el plan de calidad, el encargado de ello ayuda al equipo a llevar seguimiento.
- En el TSP, se hace mucho énfasis en el manejo de los defectos como la muestra principal de calidad.
- Una ayuda para medir la calidad se utilizan los siguientes estándares:
 - **Percent defect freeb (Porcentaje libre de defectos)**
 - **Defect – removal profile**
 - **Quality profile**
 - **Process quality index**

Ventajas del TSP

Debido al seguimiento intensivo, el proyecto tiene muchísimas más posibilidades de éxito.

Detección temprana de riesgos y defectos.

Los Ingenieros de Software aprenden como reducir los defectos de un 40 a 50%



¿Cuáles Son Las Desventajas Del TSP?



Se debe contar con un buen conjunto de métricas y parámetros de calidad.



Los miembros deben tener la disciplina para seguir el plan



Recae mucha responsabilidad en una sola persona.

REFERENCIAS

■ https://resources.sei.cmu.edu/asset_files/TechnicalReport/2000_005_001_13754.pdf

- | | | | |
|----------------------|---|---------------------------|---|
| [Cummings 87] | Cummings, Thomas G. "Self-Regulating Work Groups: A Socio-Technical Synthesis," <i>Academy of Management</i> , Vol. 3, No. 3, July 1978, p. 627. | [Humphrey 00] | Humphrey, W. S. <i>Introduction to the Team Software Process</i> . Reading, MA: Addison-Wesley, 2000. |
| [DeMarco 87] | DeMarco, T. and Lister, T. <i>Peopleware, Productive Projects and Teams</i> . New York: Dorset House Publishing Co., 1987. | [Juran 88] | Juran, J. M. and Gryna, F. <i>Juran's Quality Control Handbook, Fourth Edition</i> . New York: McGraw-Hill Book Company, 1988. |
| [Deming 82] | Deming, W. <i>Out of the Crisis</i> . , Cambridge, MA: MIT Center for Advanced Engineering Study, 1982. | [Katzenbach 93] | Katzenbach, J. and Smith, D. <i>The Wisdom of Teams</i> . Boston, MA: Harvard Business School Press, 1993. |
| [Dyer 84] | Dyer, J. "Team Research and Team Training: A State-of-the-Art Review." <i>Human Factors Review</i> , The Human Factors Society, Inc. (1984): 286, 309. | [Mohrman 95] | Mohrman S. <i>Designing Team-Based Organizations, New Forms for Knowledge Work</i> . San Francisco, CA: Jossey-Bass Publishers, 1995. |
| [Fagan 76] | Fagan, M. "Design and Code Inspections to Reduce Errors in Program Development." <i>IBM Systems Journal</i> 8, 3 (1976). | [Paulk 95] | Paulk, M. et al., <i>The Capability Maturity Model: Guidelines for Improving the Software Process</i> . Reading, MA: Addison Wesley, 1995 |
| [Fagan 86] | Fagan, M. "Advances in Software Inspections." <i>IEEE Transactions on Software Engineering SE-12</i> , 7 (July 1986). | [Shaw 81] | Shaw, M. <i>Group Dynamics: The Psychology of Small Group Behavior</i> . New York: McGraw-Hill, 1981. |
| [Ferguson 97] | Ferguson, P.; Humphrey, W. S.; Khajenoori, S.; Macke, S.; and Matvya, A. "Introducing the Personal Software Process: Three Industry Case Studies." <i>IEEE Computer</i> 30 (May 1997): 24-31. | [Shellenbarger 00] | Shellenbarger, S. "To Win the Loyalty of Your Employees, Try a Softer Touch." <i>The Wall Street Journal</i> , January 26, 2000: B1. |
| [Humphrey 89] | Humphrey, W. S. <i>Managing the Software Process</i> . Reading, MA: Addison-Wesley, 1989. | [Stevens 94] | Stevens. M. and Campion, M. "The Knowledge, Skill, and Ability Requirements for Teamwork: Implications for Human Resource Management." <i>Journal of Management</i> 20, 2 (1994). |
| [Humphrey 95] | Humphrey, W. S. <i>A Discipline for Software Engineering</i> . Reading, MA: Addison-Wesley, 1995. | [Webb 00] | Webb, D. R. "Managing Risk With TSP." <i>Crosstalk</i> , June 2000: 20. |
| [Humphrey 97] | Humphrey, W. S. <i>Introduction to the Personal Software Process</i> . Reading, MA: Addison-Wesley, 1997. | | |
| [Humphrey 98] | Humphrey, W. S. "The Software Quality Profile." <i>Software Quality Professional</i> 1, 1 (1998): 8-18. | | |