Software Engineering Institute

Team Software Process[SM] (TSP[SM]) Body of Knowledge (BOK)

Watts S. Humphrey
Timothy A. Chick
William Nichols
Marsha Pomeroy-Huff

July 2010

TECHNICAL REPORT
CMU/SEI-2010-TR-020
ESC-TR-2010-020

Software Engineering Process Management

Unlimited distribution subject to the copyright

http://www.sei.cmu.edu

Carnegie Mellon

This report was prepared for the

SEI Administrative Agent
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

Copyright 2010 Carnegie Mellon University.

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

Internal use. Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use. This document may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

For information about SEI publications, please visit the library on the SEI website (www.sei.cmu.edu/library).

Table of Contents

Table of Contents i

List of Figures v

Acknowledgments vii

Executive Summary ix

Abstract xi

1 Introduction 1

1.1 Purpose for the TSP BOK 2

1.2 Sources and Influences 3

1.3 Document Organization 3

2 TSP BOK Structure and Terminology 5

2.1 Structure of the BOK 5

2.2 Operational Definition of Terms 5

3 The TSP Body of Knowledge 7

Competency Area 1: TSP Foundations and Fundamentals 9

Knowledge Area 1.1: Knowledge Work 9

Knowledge Area 1.2: TSP Prerequisite Knowledge 12

Knowledge Area 1.3: TSP Principles 14

Knowledge Area 1.4: TSP Process Elements and Measures 15

Knowledge Area 1.5: TSP Quality Practices 17

Competency Area 2: Team Foundations 19

Knowledge Area 2.1: Teams and Teambuilding 19

Knowledge Area 2.2: Team Types, Styles, and Dynamics 22

Knowledge Area 2.3: Team Formation and Membership 26

Knowledge Area 2.4: Team Member Responsibilities 28

Knowledge Area 2.5: Team Member Roles 29

Knowledge Area 2.6: Team Leader Role 34

Knowledge Area 2.7: Coach Role 37

Competency Area 3: Project Planning with TSP 41
Knowledge Area 3.1: Change Management Fundamentals 41
Knowledge Area 3.2: Piloting TSP in an Organization 45
Knowledge Area 3.3: Preparing Management and Teams for TSP Implementation 48
Knowledge Area 3.4: The TSP Launch Meetings 51
Knowledge Area 3.5: The TSP Relaunch 57
Competency Area 4: Project Implementation and Tracking with TSP 61
Knowledge Area 4.1: Weekly Meetings 62
Knowledge Area 4.2: Checkpoints 64
Knowledge Area 4.3: Communicating with Stakeholders 65
Knowledge Area 4.4: Replanning 67
Knowledge Area 4.5: Phase, Cycle, and Project Postmortems 70
Competency Area 5: Gathering and Using TSP Data 72
Knowledge Area 5.1: Data Recording 73
Knowledge Area 5.2: Gathering and Using Size Data 73
Knowledge Area 5.3: Gathering and Using Schedule Data 75
Knowledge Area 5.4: Gathering and Using Quality Data 75
Knowledge Area 5.5: Gathering and Analyzing Postmortem Data 79
Competency Area 6: Scaling Up the TSP 82
Knowledge Area 6.1: Organizational Implementation 82
Knowledge Area 6.2: TSP Process Variations 84
Knowledge Area 6.3: Large-scale TSP Teams 90
Appendix A: Engineering Guidelines 92
A1 Principles of Modern Engineering Work 92
Appendix B: Project Management Guidelines 94
B1 Operational Processes for Project Management 94
B2 Project Management Using TSP 94
B3 Managing TSP Plans 96
B4 Managing Team Communication 97
B5 Managing Team Project Focus 97
B6 Managing Team Roles 98

Appendix C: TSP Coaching Guidelines 104

C1 The TSP Coach Role 104

C2 Guidelines for Introducing TSP into an Organization 105

C3 Guidelines for Launching Teams 107

C4 Guidelines for Coaching Teams 109

C5 Guidelines for Coaching Role Managers 111

C6 Guidelines for Assessing Team Characteristics 112

C7 Guidelines for Coaching Plan Management Issues 114

C8 Guidelines for Coaching Data Management Issues 116

C9 Guidelines for Data Analyses 118

C10 Guidelines for Coaching the Team's Quality Management 120

C11 Guidelines for Coaching the Team's Schedule Tracking 120

C12 Guidelines for Coaching the Postmortem 122

C13 Guidelines for Coaching TSP Multi-teams (TSPm) 122

C14 Guidelines for Coaching Other TSP Team Types 124

Appendix D: TSP Team Leader Guidelines 126

D1 The Team Leader Role 126

D2 Team Leader Guidelines for Plan Management 127

D3 Team Leader Guidelines for Quality Management 128

D4 Developing the Team 128

D5 Protecting the Team 129

D6 Working with the TSP Coach 129

Acronyms Used 131

References 132

List of Figures

Figure 1: Architectural Hierarchy of the TSP BOK Components 5

Figure 2: Group Formation Stages 21

Figure 3: Transactional and Transformational Leadership Styles 23

Figure 4: Traditional and Self-Directed Teams 24

Figure 5: Team Working Styles 25

Figure 6: The Conner-Patterson Model of Technology Adoption in Organizations 43

Figure 7: Adopter Categories as Defined by Everett Rogers 44

Figure 8: The TSP Launch Meeting Sequence 52

Figure 9: The TSP Relaunch Meeting Sequence 59

Acknowledgments

In preparing this report, the authors consulted with several individuals who provided ideas and contributions to the content of the TSP BOK. In particular, we want to acknowledge those that contributed to reviewing the content and clarity of the report, and we thank these individuals for their time and assistance: Yoshi Akiyama, Olivia Barron, Daniel Burton, Lana Cagle, Ahmed El-Shikh, Bradley Hodgins, James McHale, Said Nurhan, James Over, and Alan Willett.

Executive Summary

As the character of engineering technology has changed in the post-industrial revolution, an increasing proportion of engineered products are actually components of entire systems of products that directly support end-use applications such as driving, flying, or medical diagnoses and treatments. These products and systems must meet critical performance, safety, security, survivability, and usability requirements. Not only must these modern engineering products be of the highest possible quality, but they also must meet business-critical schedule and budget constraints.

Modern engineering work requires teams for work products that are too large or too complex to be completed by a single engineer. Furthermore, the modern engineering workforce must work in close cooperation with people who have the variety of domain skills required for the system's design and implementation. This requires a work environment in which people with vastly different skills can work together to produce quality products that meet their functional, architectural, and property requirements. The Personal Software Process^{SM} (PSP^{SM}) and Team Software Process^{SM} (TSP^{SM}) technologies provide such an environment by proving individuals and teams with a framework for creating or tailoring processes that all members can follow, for communicating in a common vocabulary, and for planning and tracking their work using a commonly accepted set of measurements and standards.

The Team Software Process Body of Knowledge (TSP BOK) was drafted to define the fundamental knowledge and skills that set TSP-trained individuals apart from other software professionals. It helps individual practitioners to assess and improve their own skills, provides employers with an objective baseline for assessing the process improvement skills and capabilities of their development team members, and guides academic institutions that want to incorporate TSP into their software and other engineering courses or curricula. The TSP BOK also facilitates the development of TSP certification programs that are based on a well-established standard set of knowledge and skills.

The TSP BOK is intended to provide a high-level comprehensive overview of the competencies that compose the essential knowledge and skills required for the competent implementation of the TSP as a team member, team leader, coach, or manager of a TSP team. This document is not meant to provide detailed descriptions or in-depth explanations of the concepts, practices, and procedures of every component in the TSP. Rather, the purpose of this document is to provide an overview of the competencies, knowledge areas, and key concepts and skills that constitute the essential knowledge, skills, and abilities of competent TSP practitioners.

Abstract

The Team Software Process Body of Knowledge (TSP BOK) was drafted to define the fundamental knowledge and skills that set TSP-trained individuals apart from other software professionals. It helps individual practitioners to assess and improve their own skills, provides employers with an objective baseline for assessing the process improvement skills and capabilities of their development team members, and guides academic institutions that want to incorporate TSP into their software and other engineering courses or curricula. The TSP BOK also facilitates the development of TSP certification programs that are based on a well-established standard set of knowledge and skills.

1 Introduction

As the character of engineering technology has changed in the post-industrial revolution, an increasing proportion of engineered products are actually components of entire systems of products that directly support end-use applications such as driving, flying, or medical diagnoses and treatments. These products and systems must meet critical performance, safety, security, survivability, and usability requirements. Not only must these modern engineering products be of the highest possible quality, but they also must meet business-critical schedule and budget constraints.

Modern engineering work requires teams for work products that are too large or too complex to be completed by a single engineer. Furthermore, the modern engineering workforce must work in close cooperation with people who have the variety of domain skills required for the system's design and implementation. This requires a work environment in which people with vastly different skills can work together to produce quality products that meet their functional, architectural, and property requirements. The Personal Software Process^{SM} (PSP^{SM}) and Team Software Process^{SM} (TSP^{SM}) technologies provide such an environment by proving individuals and teams with a framework for creating or tailoring processes that all members can follow, for communicating in a common vocabulary, and for planning and tracking their work using a commonly accepted set of measurements and standards.

The PSP is a disciplined and structured approach to developing software that was developed in 1993 by Watts S. Humphrey [Humphrey 1995]. By using the PSP concepts and methods in their work, individuals in almost any technical field can improve their estimating and planning skills, make commitments that they can meet, manage the quality of their work, and reduce the number of defects in their products. The TSP was introduced in 1998, and builds upon the foundation of PSP to enable engineering teams to build software-intensive products more predictably and effectively [McAndrews 2000].

The PSP and TSP technologies are based on the premise that a defined and structured process can improve individual work quality and efficiency. A process that requires professionals to define, measure, and track their work can help them to better understand what they do, and provide them with the necessary information to evaluate and learn from their experiences. By developing their own defined processes within the PSP framework, professionals gain the knowledge and experience needed to select the methods and practices that are best suited to their particular tasks and abilities. Similarly, the TSP provides teams who are building software-intensive products or systems with a framework in which individuals can combine their personal process discipline skills with proven process management techniques that enable them to do high-quality work. The use of agreed-upon, well-defined processes also provides teams with a foundation for effective collaboration and an environment that facilitates creative and productive work.

Implementation of the TSP begins with a launch process in which a coach guides the team and their managers through the steps of establishing goals, defining team roles, producing a team plan, assessing risks and devising possible mitigations, and obtaining management approval for the

project plan. Following the launch, the team uses the TSP process framework for managing, tracking, and reporting the team's progress against its cost, schedule, and product quality goals. The concepts and methodologies of the PSP and TSP technologies have reached a level of maturity sufficient to warrant the development of professional competency measures to assess both the level of knowledge acquisition and the level of skill in applying that knowledge. At the core of the process of maturing a profession is the establishment of a *body of knowledge* (BOK).

A body of knowledge is a document generated by master practitioners in a particular profession to identify and delineate the concepts, facts, and skills that competent professionals in that field are expected to have mastered. The Institute of Electrical and Electronics Engineers (IEEE) Computer Society has established a body of knowledge for the software engineering profession as a whole [IEEE 2004]. The TSP BOK document is meant to complement and build upon that work and upon the PSP BOK [Pomeroy-Huff 2009] by describing the essential skills and knowledge specific to the TSP methodology for software process improvement.

1.1 Purpose for the TSP BOK

The TSP BOK was drafted to define the fundamental knowledge and skills that set TSP-trained individuals apart from other software professionals. It helps individual practitioners to assess and improve their own skills, provides employers with an objective baseline for assessing the process improvement skills and capabilities of their development team members, and guides academic institutions that want to incorporate TSP into their software and other engineering courses or curricula. The TSP BOK also facilitates the development of TSP certification programs that are based on a well-established standard set of knowledge and skills.

The TSP BOK is intended to provide a high-level comprehensive overview of the competencies that compose the essential knowledge and skills required for the competent implementation of the TSP as a team member, team leader, coach, or manager of a TSP team. This document is not meant to provide detailed descriptions or in-depth explanations of the concepts, practices, and procedures of every component in the TSP. Rather, the purpose of this document is to provide an overview of the competencies, knowledge areas, and key concepts and skills that constitute the essential knowledge, skills, and abilities of competent TSP practitioners. The main purposes of this document are as follows.

1. Define the essential knowledge and skills that TSP-trained professionals are expected to master
2. Characterize the standard practices of TSP-trained professionals
3. Delineate the knowledge and skills that set TSP practitioners apart from ordinary engineering professionals
4. Establish a baseline for developing, assessing, and accrediting TSP courses and curricula throughout academia
5. Facilitate the establishment of TSP certification programs that are based on an established and agreed-upon standard knowledge and skills set
6. Provide employers with a baseline for assessing the skills and capabilities of their product development team members to identify those areas in which additional training might be required
7. Characterize the disciplined practices used by self-directed TSP team members

Another purpose of this document is to define and delineate the baseline knowledge and skill set upon which the Carnegie Mellon® Software Engineering Institute (SEI) certification program for the SEI-Certified TSP Coach is based.

Although the TSP BOK is meant to guide the design, development, implementation, and assessment of courses and curricula based in part or in whole on the knowledge and skills delineated in it, the TSP BOK is not intended to be a guide for curriculum or course development. Such activities require pedagogical knowledge and expertise outside the domain of this body of knowledge; therefore, this document is meant to guide only the content – not the methodology – of TSP instruction and training.

1.2 Sources and Influences

In preparing this document, the authors examined a number of reports delineating bodies of knowledge from other professional disciplines. Of these, three body of knowledge guidebooks provided guidance and inspiration in terms of structuring the document and depicting the architectural hierarchy used to describe the TSP BOK. These guides were

- IEEE Computer Society's *Guide to the Software Engineering Body of Knowledge (SWEBOK)*, 2004 Version
- Project Management Institute's *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*, Fourth Edition
- SEI's *Personal Software Process Body of Knowledge (PSP BOK)*, Version 2.0 (CMU/SEI-2009-SR-018)

The IEEE *SWEBOK* 2004 Version and *PMBOK Guide* were influential in determining the document flow and delineation of components used in the description of the TSP BOK. The PSP BOK defines the foundational set of prerequisites, on which rest the additional knowledge and skills needed for effective practice of the TSP methodologies.

1.3 Document Organization

This document is composed of six major sections.

1. Section 1 (this portion of the TSP BOK) provides background information, an overview of the intended purposes of and audience for this body of knowledge, and the sources and influences that affected its development.
2. Section 2 summarizes the structure of the hierarchy used to describe the content of the body of knowledge and provides operational definitions of terms used in the document.
3. Section 3 outlines the competency and knowledge areas that make up the body of knowledge and delineates the key concepts and skills that make up each knowledge area.
4. The fourth portion of the document contains appendices which provide guidelines and suggestions to TSP team members, project managers, coaches, and team leaders.
5. The fifth section provides a list of acronyms commonly used in TSP (and PSP).
6. The last section contains complete citations for works referenced in this document.

®Carnegie Mellon is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

2 TSP BOK Structure and Terminology

2.1 Structure of the BOK

As with other professional body of knowledge documents, the information contained in the TSP BOK is organized into competency areas, each of which is composed of a group of interrelated knowledge areas. The knowledge areas, in turn, are composed of concepts and skills, which are the smallest units of information contained in the body of knowledge. For the purpose of this model, the term *concept* is used to describe the intellectual aspects of the TSP content; that is, the information, facts, terminology, and philosophical components of the technology. The term *skill* refers to the ability of an individual to interpret and apply one or more concepts to the performance of a task. In this document, it is assumed that if individuals understand a concept, then they also have the ability to perform the skills related to or founded upon that concept.

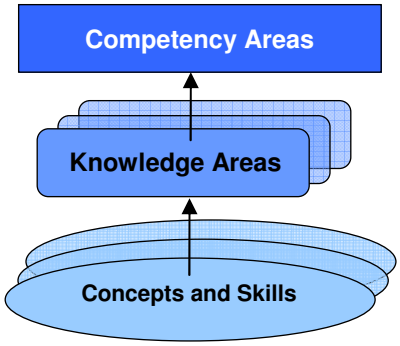

Figure 1: Architectural Hierarchy of the TSP BOK Components

2.2 Operational Definition of Terms

The TSP BOK uses the following terms to describe the categories of principles and processes it contains.

Competency area A group of closely-related knowledge areas that a practitioner is well qualified to perform intellectually or physically

Knowledge area The sum or range of specific understanding and ability gained through study of a set of concepts or through experience with a set of skills

Concept An explanatory principle applicable to a specific instance or occurrence within a particular knowledge area

Skill Proficiency, facility, or dexterity of performance that is acquired or developed through training or experience in a particular knowledge are

Engineering terms used in the TSP BOK are operationally defined as follows.

Design The specification and plan for the subsequent implementation or manufacture of some item

Design process A standardized set of actions required to produce a design

Developer A person who designs software and writes the programs; or, an organization that designs software and produces code as its primary business function

Development The process of elaborating a design into a sufficiently detailed form so that it can be used to produce a quality product that conforms to the intent of the designers, as well as the creation of the designed product or system. *Software development* includes the design of the user interface and the program architecture, as well as programming the source code. [http://www.pcmag.com/encyclopedia]

Engineer A person who has been trained for and is engaged in professional engineering work

Engineering The application of scientific and mathematical principles and methods to the design, manufacture, operation, and support of structures, machines, products, and systems

Planning horizon The portion of the project for which a detailed plan exists. (In this context, "detailed plan" means that the project work has been broken into a logical sequence of tasks or sub-tasks, with each task or sub-task requiring no more than 8 to 10 hours for completion.) The length of a planning horizon varies according to the size of the project: for small or short-term projects, the planning horizon could include the entire scope of the work, whereas the planning horizon of large or long-term projects covers only the portion of the work that will be completed in the several weeks following a launch or relaunch, or during a particular project phase or cycle.

Professional A person who is engaged in an occupation or is a member of a vocation that requires specialized education and/or training. Professionals are expected to conform to the technical and ethical standards of the discipline in which they have attained professional status.

Quality The measurement of the degree or level of excellence of a engineered system, product, or service

3 The TSP Body of Knowledge

The TSP BOK is composed of six competency areas, each with several knowledge areas.

Competency Area 1: TSP Foundations and Fundamentals

1.1 Knowledge Work
1.2 TSP Prerequisite Knowledge
1.3 TSP Principles
1.4 TSP Process Elements and Measures
1.5 TSP Quality Practices

Competency Area 2: Team Foundations

2.1 Teams and Teambuilding
2.2 Team Types, Styles, and Dynamics
2.3 Team Formation and Membership
2.4 Team Member Responsibilities
2.5 Team Member Roles
2.6 Team Leader Role
2.7 Coach Role

Competency Area 3: Project Planning with TSP

3.1 Change Management Fundamentals
3.2 Piloting TSP in an Organization
3.3 Preparing Management and Teams for TSP Implementation
3.4 The TSP Launch Meetings
3.5 The TSP Relaunch

Competency Area 4: Project Implementation and Tracking with TSP

4.1 Weekly Meetings
4.2 Checkpoints
4.3 Communicating with Stakeholders
4.4 Replanning
4.5 Phase, Cycle, and Project Postmortems

Competency Area 5: Gathering and Using TSP Data

5.1 Data Recording
5.2 Gathering and using Size Data
5.3 Gathering and Using Schedule Data
5.4 Gathering and Using Quality Data
5.5 Gathering and Analyzing Postmortem Data

Competency Area 6: Scaling Up the TSP

6.1 Organizational Implementation
6.2 TSP Process Variations
6.3 Large-scale TSP Teams

The remainder of this section contains a description of each competency area, its supporting knowledge areas, and the key concepts composing each knowledge area. This information is not meant to provide detailed descriptions of every concept, element, or practice that makes up the Team Software Process methodology; rather, this BOK is meant to provide a high-level overview of the areas in which a competent practitioner of the TSP is expected to be proficient.

Competency Area 1: TSP Foundations and Fundamentals

This competency area outlines the foundational knowledge on which TSP is built and describes the fundamental concepts that a TSP practitioner must understand in order to successfully implement and practice the TSP methodology. The knowledge areas composing the TSP Foundations and Fundamentals competency area are as follows.

1.1 Knowledge Work – PSP and TSP are practices designed to facilitate and improve both the process and the outputs of knowledge work, which is the interpretation, development, and implementation of information by skilled professionals within a specific subject area. This knowledge area discusses the nature of knowledge work and the team and workplace characteristics required for such work.

1.2 TSP Prerequisite Knowledge – This knowledge area outlines the fundamental concepts and skills that individuals must master before implementing the TSP methodology as a member of a TSP team. Although this area calls out some of the specific Knowledge Areas of the PSP BOK, the PSP BOK in its entirety is considered to be prerequisite knowledge for implementing the TSP in practice.

1.3 TSP Principles – This knowledge area outlines the basic principles underlying the Team Software Process. The key concepts identify the elements that are common to and required for successful outcomes of work done by teams to produce software products and/or software-intensive systems.

1.4 TSP Process Elements and Measures – This knowledge area describes the process elements and measures that are used in the TSP. (Where applicable, overlaps with or differences from PSP process elements and measures are noted.)

1.5 TSP Quality Practices – This knowledge area describes the specific quality practices added in the TSP to build on the individual quality practices used by PSP practitioners.

References: The material covered in this competency area is detailed in these primary sources.
[Humphrey 1995, Chapters 1, 11, Appendix A, Appendix C]
[Humphrey 1999, Chapters 1 and 6]
[Humphrey 2005, Chapters 2, 6, 13]
[Pomeroy-Huff 2009, in its entirety]

Knowledge Area 1.1: Knowledge Work

PSP and TSP are practices designed to facilitate and improve both the process and the outputs of knowledge work, which is the interpretation, development, and implementation of information by skilled professionals within a specific subject area [Drucker 1999]. This knowledge area discusses the nature of knowledge work and the team and workplace characteristics required for such work.

1.1.1 Characteristics of knowledge work

Software and systems engineering work is considered to be knowledge work because the nature of the tasks required to create the final products is largely intellectual. Knowledge work differs from traditional product development work in the following ways.

- The quality, cost, and schedule performance of such work is largely determined by the motivation, skill, and discipline of the workers, rather than the cost of the raw materials and labor or the efficiency of manual work processes.
- Knowledge work consists largely of converting information from one form to another; therefore, the results of a knowledge work process are frequently intangible and may therefore be hard to measure or assess [Kidd 1994].
- The suitability of knowledge workers' products generally cannot be determined except through extensive final use or by rigorous examination by other adept and knowledgeable knowledge workers.

1.1.2 Characteristics of knowledge workers

The fundamental characteristic that sets knowledge workers apart from other types of production workers is ownership: in traditional manufacturing, the employer (or organization) owns both the assets required to produce the end product and the product itself; in knowledge work, the workers own the assets and retain at least partial ownership of the end product, even if the product is considered the "intellectual property" of the employer [Drucker 1999].

Other characteristics unique to knowledge workers are as follows.

- All knowledge workers (even those in the same field) are different from one another, so no single generic process can be developed to manage all knowledge work. For maximum effectiveness, processes for knowledge work must be customized to suit – and preferably, should be generated by – the people who are going to implement those processes [Davenport 2005a].
- Knowledge workers are highly motivated to learn, and the act of learning changes the way that they think. The very act of working causes knowledge workers to change their own "internal configurations" and enhances their skill sets and working abilities [Kidd 1994]. These workers view themselves as "assets" to the company, and if companies are to retain knowledge workers, they too must view theses workers as assets, rather than as costs [Drucker 1999].
- Knowledge workers need autonomy; they don't like to be told what to do and how to do it [Davenport 2005b]. Because of the degree of education and experience needed to do knowledge work, knowledge workers are used to working without constant and stringent supervision – knowledge workers' creativity is actually stifled by micro-management. Knowledge workers produce the best results when they are allowed to formulate their own processes for doing their work.
- It is more difficult and less valuable to specify a detailed sequence of steps for knowledge work than for other types of work [Davenport 2005b]. The workflow of knowledge work generally varies widely among individuals engaged in the same type of work; further, the steps and phases tend to vary widely from project to project – and as the complexity of the work increases, so does the variation in the workflow. It is possible to apply a process perspective to improving knowledge work, but there is no way to formulate a "one-size-fits-all" approach to process improvement. Such improvements must be tailored to fit the work and should emphasize a disciplined approach to the work, rather than rigid adherence to specific workflow structures.

- Productivity and outputs of knowledge workers are difficult to quantify or to specify in detail. Often, the work process itself is largely invisible to observers: it is difficult to tell when someone is thinking (or not), and trying to quantify knowledge work outputs or productivity is difficult and often meaningless ("ideas per hour") for the purpose of assessing or improving performance. Knowledge work outputs must be assessed in terms of product quality or goodness-of-fit of the result with its intended implementation as a solution to a problem.
- Commitment affects knowledge workers' performance [Davenport 2005b]. The performance of knowledge workers is directly correlated with their degree of mental and emotional commitment to the work. The best way to establish and maintain a high degree of commitment in knowledge workers is to make sure that they understand what they are being asked to do and why it is significant, and to give the workers some say in what they do and how they do it.

1.1.3 Characteristics of effective knowledge worker management

Knowledge workers must be effectively managed and provided with a supportive environment if they are to consistently and predictably produce high-quality knowledge work.

- Knowledge workers themselves are the only ones who can understand how their work is done. Therefore, management cannot manage knowledge work; rather, knowledge workers must manage it themselves. When knowledge workers manage their own work, they tend to be more productive than when someone else attempts to manage them [Drucker 1999].
- For knowledge workers to effectively manage themselves, they must know how to manage, and they must be motivated both to learn and to implement effective self-management practices [Humphrey 2006c].
- The principal areas of self-management on which knowledge workers must focus include
  - planning and tracking the work
  - resolving problems and issues that will adversely affect the work (at both individual and team levels)
  - identifying and managing risks to the work
  - controlling work time and striving to improve productivity
  - staying focused on the highest priority tasks

1.1.4 Factors governing knowledge worker productivity

There are six primary factors that affect knowledge worker productivity [Drucker 1999].

1. Knowledge worker productivity demands that the knowledge workers clearly understand the task at hand.
2. The responsibility for knowledge workers' productivity must be imposed on the individuals. Knowledge workers have to manage themselves.
3. Continuing innovation has to be part of the work, the task, and the responsibility of knowledge workers.
4. Knowledge work requires continuous learning on the part of the knowledge worker, and equally, continuous teaching on the part of the knowledge worker.
5. The primary productivity measure of the knowledge worker should not be a matter of quantity of output. Quality is at least as important as quantity.

6. Knowledge worker productivity requires that the knowledge worker is both seen and treated as an asset rather than as a cost. The most productive knowledge worker is one who wants to work for the organization in preference to all other opportunities.

1.1.5 How TSP enables and enhances knowledge worker productivity

In the modern workplace, most knowledge work is carried out by teams, rather than individuals. The same factors that govern individual knowledge workers also apply to teams of knowledge workers, although additional support may be needed to scale knowledge work up to the team level. The TSP launch process builds teams of knowledge workers who have the skill, motivation, opportunity, and support to manage themselves.

Knowledge Area 1.2: TSP Prerequisite Knowledge

The TSP is an extension of the PSP, and builds upon the complete body of knowledge set forth for PSP practitioners. Although the PSP BOK in its entirety is considered to be prerequisite knowledge for TSP practitioners, this knowledge area calls out the basic fundamental concepts and skills that individuals must master before implementing the TSP methodology as a member of a TSP team.

1.2.1 Defining and using processes

A *process* describes the sequence of steps that a knowledgeable professional should follow when doing a specified task. A *defined process* is a documented set of steps or activities for doing a specific job or task. The PSP is a series of defined processes that allow individuals to produce high-quality products on time and within budget. Within these defined processes are sets or series of elements or steps called *phases*. Whatever its structure, a process must be utilized and followed as defined if it is to produce reliable outcomes and meaningful data. The fundamental concepts and skills relevant to defining and using processes in the PSP can be found in Knowledge Areas 1.1 and 2.1of the PSP BOK.

1.2.2 Collecting data for process improvement

The concept of process improvement in general – with PSP and TSP as process improvement methodologies – is based on the realization that individuals must understand precisely how they work and what they do before they can improve their personal performance and produce quality products on predictable schedules. Therefore, individuals should begin any process improvement effort by collecting data; only when individuals have data can they fully understand the nature of their work. Analyzing that data allows individuals to understand what they do and how they do it, and provides a baseline for measuring the effects of the changes implemented to improve their usual work processes and products.

1.2.3 Analyzing data for process improvement

Individuals should analyze their personal process data to learn about the manner in which they work, to assess the quality of the products that they produce and of the process used to produce those products, and to help establish realistic performance goals. To improve a process, that process must be used consistently and produce consistent results; therefore, the variability of the process and outputs should be understood statistically.

Because the results of a superficial analysis can be misleading, data analyses should be conducted using statistically sound methods. Although individual data points can provide a rough snapshot of progress or performance, process analysis should be based on multiple data points. The greater the number of data points available for analysis, the more precise the information is likely to be, which reduces the likelihood that chance variation is mistaken for change in the process output.

Personal data analysis provides a variety of information that can enable individuals to

- learn about their products and processes
- establish standards and specifications for their products and processes
- determine if some product or process meets its defined criteria
- precisely control what they do
- develop indicators of their performance
- improve their personal performance
- manage the quality of the products they produce
- estimate when they will finish a task
- estimate expected performance ranges
- precisely plan, track, and report on their work

1.2.4 Using statistical foundations for process improvement

Although individual data points can provide a rough indication of progress or performance, it is easy to be misled by superficial data analyses. For precise information and control, it is essential to collect multiple data points and to analyze the data using statistically sound methods.

Statistics are the foundation for project planning and tracking using PSP and TSP, and provide an objective means for analyzing and improving personal and team performance. TSP team members should have an appreciation of the following basic statistical topics.

- Arithmetic mean
- Statistical distributions
- Variance and standard deviation
- Correlations and their significance
- Linear regression, multiple regression, and prediction intervals
- Pareto distributions
- Numerical integration
- Tests for normality
- Gauss's method

A complete annotated list of the principle statistics used in PSP (and to some extent, in TSP) can be found in the PSP BOK, Knowledge Area 1.3 and Appendix A.

1.2.5 Other foundations for TSP

The PSP BOK in its entirety is considered to be fundamental foundational knowledge for successfully implementing the TSP in practice.

Knowledge Area 1.3: TSP Principles

This knowledge area outlines the basic principles underlying the Team Software Process. The key concepts identify the elements that are common to and required for successful outcomes of work done by teams to produce software products and/or software-intensive systems.

1.3.1 Use structured processes

Knowledge work is most effective when it is done using a defined, structured process with repeatable and measurable steps that provide workers with rapid feedback on the quality of the product and progress towards completion.

1.3.2 Establish a shared understanding

Productive teamwork requires that the team have a shared understanding what the work entails and how it is to be done. Each member needs to have the same understanding of the team goals, team member's roles, products or components to be produced, available resources and existing constraints, and measures of success.

1.3.3 Use proven techniques

Knowledge work, whether done by individuals or teams, is most productive when it is based on sound professional practices and proven techniques as relevant to the desired outcome. TSP provides a framework and guidelines that are based on proven best practices for organizational oversight, team management, personal discipline, and process improvement at the individual, team, project, and enterprise levels.

1.3.4 Use data whenever and wherever possible

Plans, estimates, process improvement decisions, quality ratings, and all other aspects of the development process should be based on data whenever possible. To get useful data for future projects, data on current projects should be carefully and accurately gathered and recorded as close as possible to the actual time of the data generation. The best way to ensure data accuracy is to treat data as the private property of the individual whose work is being measured. If individuals or teams believe that management will attempt to rate their performance based on personal data, it is unlikely that unfavorable results will be recorded accurately, if at all – in which case, the data will become useless for planning and tracking purposes.

1.3.5 Make realistic plans and commitments

The only realistic plans and commitments are those that are made by the individual(s) responsible for meeting them: team plans and team commitments must be made by the teams who will do the work. Any commitment should be based on available data, realistic evaluation of available resources, the nature of the work, and the skills and abilities of the team. All of this information should be used to generate a plan that will determine whether or not the proposed work can be produced within the desired cost and schedule parameters. Only if the plan indicates that the work is feasible should a commitment be negotiated.

1.3.6 Practice self-management

Knowledge workers are best enabled to do creative work when they manage themselves. This means that individuals or teams must be allowed to negotiate their own commitments, make their

own plans, follow their own chosen processes for doing the work, choose their own standards and methods for ensuring good product quality, identify and mitigate risks to the work, maintain open and regular communication with other team members, management, and stakeholders, and take responsibility for meeting their planned goals and commitments.

1.3.7 Focus on quality

The primary focus of TSP team members is on producing high quality components and products. Defective products are expensive because finding and fixing problems is both time-consuming and resource-intensive; the cost of time and resources increases exponentially when defects are discovered later rather than sooner in the process. It is faster and cheaper to build a product right than to build a product (or components of the product) over. Therefore, focusing on quality – of both the product and the methods and processes used to produce it – allows teams to economically and predictably produce a superior product [Humphrey 2006a].

1.3.8 Regard design as a fundamental element of quality work

A comprehensive and detailed design is one of the single most effective defect-prevention techniques available to developers. Even when the best implementation practices are used, the quality of the resultant products is only as good as the quality of the design methods used. TSP methods for ensuring that design is a fundamental part of the work process include dedicated process phases for design, the designated team member role for design management, inclusion of guidelines and checklists for producing and reviewing designs, and scripted activities for ensuring that the team adopts and uses sound design methods and standards.

Knowledge Area 1.4: TSP Process Elements and Measures

This knowledge area describes the process elements and measures that are used in the TSP. (Where applicable, overlaps with or differences from PSP process elements and measures are noted.)

1.4.1 Defined processes in TSP

The TSP encompasses two types of defined processes.

- *Personal processes* are defined sets of steps or activities that guide individuals in doing personal work. It is usually based on personal experience. A personal process can be a wholly new creation, or a modification of an established process.
- *Team processes* are sets of defined steps that each team member follows in the same way when performing the team's work. Team processes for building components and products are usually based on the collective experiences of the team members, and typically are redefined for or tailored to address the challenges faced by the team when beginning a new project. Other processes, such as the team launch (see Knowledge Area 3.4) and the weekly team meetings (see Knowledge Area 4.1) are defined as part of the TSP framework and initially are used "as-is," although they may later be modified to better fit the team's needs as the members gain more experience in using TSP.

1.4.2 TSP process phases

The TSP process provides guidance and process examples; TSP teams define their own project processes during the team launch. This enables TSP teams to tailor their processes to the special needs of their projects, organizations, and teams.

1.4.3 TSP process elements

Process elements are components of a process. The TSP uses seven basic process elements: the five basic process elements found in PSP (scripts, forms, measures, checklists, and standards), plus specifications and guidelines. As in PSP, there are four basic TSP data measures: time, size, quality (defects), and schedule. Process elements used by both PSP and TSP are explained in detail in Knowledge Area 1.2 of the PSP BOK; TSP specifications and guidelines are explained below in 1.4.5 and 1.4.6.

1.4.4 Basic data collection measures

The PSP has four basic measures: time, size, quality (defects), and schedule. These parameters are used for making estimates based on actual data (or PSP guidelines, in the absence of historical data) and for collecting data that can be used for making future estimates. The PSP measures are explained in Knowledge Area 2.2 of the PSP BOK. Measures specific to TSP are described below in sections 1.4.7 through 1.4.10.

1.4.5 Specifications

Specifications are process elements that are used in TSP but not PSP. Specifications (such as the Project Status Report and the various Role Specifications) provide clear and unambiguous descriptions of a product, artifact, or task, and the criteria by which the outputs should be evaluated. The precise and consistent definitions provided by specifications help to guide the work and facilitate gathering and using data, thereby enabling consistency across TSP teams and between TSP teams and management.

1.4.6 Guidelines

Guidelines are process elements found in TSP that are used to outline the recommended rules or strategies that should be followed in determining a course of action.

1.4.7 TSP time measures

As in PSP, the TSP time measure is minutes. Time is tracked while doing the work. The six basic components of time data are start time, start date, end time, end date, interrupt time, and delta time. TSP time data are used to determine how much time is spent on each project effort or task and to provide information for making better time estimates on future tasks of a similar nature. The basic time measure components are fully explained in Knowledge Area 2.2 of the PSP BOK.

1.4.8 TSP task hours

Task hours is a measure of the actual time that team members spend on scheduled project tasks. The measure is calculated by subtracting the total interrupt time from the time elapsed between the start time and end time of a task, and is recorded on the TSP time log as delta time (see PSP BOK Knowledge Area 2.2). Task hours are tracked because this is the only time that actually

contributes to the project's planned tasks; accurate schedule tracking requires the removal of any time spent on off-task activities (see 1.4.9 below).

1.4.9 Off-task time

The time spent doing things other than planned project tasks generally is referred to as *off-task time.* It is not measured or tracked since it does not contribute to meeting the stated schedule goals. Off-task time includes time spent in management and administrative meetings, attending training classes, reading email, or any of the other essential activities that a team member must do. Off-task time for a given task or work period is calculated by subtracting the total delta time from the total elapsed time spent on a task.

1.4.10 TSP quality measures

The explicit TSP quality measures are the numbers of defects injected and removed by phase. Derived quality measures can be calculated once the team has a complete estimated plan or actual data for the size and time of all the products and phases for which quality measures are required.

Knowledge Area 1.5: TSP Quality Practices

This knowledge area describes the quality specific practices added in the TSP to build on the individual quality practices used by PSP practitioners.

1.5.1 Personal reviews

The *personal review* is fundamental to PSP practice. Personal reviews are conducted by individuals who examine their own products with the goal of finding and fixing as many defects as possible. Personal reviews should precede any other activity that uses the product (coding, compiling, testing, inspecting, etc.) and should be part of any TSP team activity, since individual components are used to produce the overall team product.

1.5.2 Review checklists

Review checklists are specific tools that are used when performing a quality check, and are customized to fit the needs of the individual. They are also tailored to fit specific process steps or phases. Each review checklist is customized to include the defect categories that have caused problems in the past so that these defects are checked for during the review. Checklists are useful in conducting all types of reviews and inspections.

1.5.3 Inspections

Inspections are structured team reviews of an artifact, component, or product, and are conducted in order to identify problems in the product. Inspections are usually conducted by the owner of the item(s) being inspected and two or more peers. The inspection meeting follows a defined procedure and the participants have established roles. Properly run inspections do not discuss the problems that are identified, nor does the team make any attempt to solve the problems or fix the defects. Any problems or defects found are returned to the individual who produced the artifact, since fixing defects is the responsibility of the person who produced the defects. In the TSP, inspections are conducted using the INS (Inspection) script.

1.5.4 Walkthroughs

Walkthroughs are similar to inspections in that they are quality checks conducted by a group of people in order to identify problems in a component or product, but they are less formal than inspections. A product, such as a design or code segment, is presented to an audience that raises issues and asks questions.

Competency Area 2: Team Foundations

A *team* consists of a group of people who act in cooperation to achieve a common purpose. When teams are effective in achieving their goals, it is because they are composed of members with complementary skills that work together to create a synergistic effort; effective teams achieve a kind of gestalt, in which the members' strengths are maximized and the weaknesses minimized so that the team as a whole becomes greater than the sum of its parts. Teams are especially appropriate for doing work of a highly complex nature (such as knowledge work) and for accomplishing large-scale tasks with many interdependent subtasks.

The knowledge areas composing the Team Foundations competency area are as follows.

2.1 Teams and Teambuilding – This knowledge area describes the characteristics of teams and explains concepts for building high-performing project teams.

2.2 Team Types, Styles, and Dynamics – This knowledge area describes some of the models of team types, team styles, and team dynamics that provide a foundational understanding of team work in the TSP.

2.3 Team Formation and Membership – This knowledge area describes important parameters that should be considered when forming and populating TSP teams.

2.4 Team Member Responsibilities – This knowledge area describes the team's specific responsibilities for helping teams to be fully effective.

2.5 Team Member Roles – This knowledge area discusses the types of roles on TSP teams and delineates the responsibilities required of the various team member roles.

2.6 Team Leader Role – This knowledge area lists the roles that a TSP team leader must fulfill and discusses some of the tasks and responsibilities that accompany the various roles.

2.7 Coach Role – This knowledge area describes the general roles and responsibilities of the TSP coach.

References: The material covered in this competency area is detailed in these primary sources.
[Humphrey 1999, Chapter 17]
[Humphrey 2006a, Chapters 1, 2, 3, 4, 8]
[Humphrey 2006b, Chapters 1, 4, 5, 6]

Knowledge Area 2.1: Teams and Teambuilding

This knowledge area describes the characteristics of teams and explains concepts for building high-performing project teams.

2.1.1 Teams

As defined by the TSP, a *team* consists of at least three people who share a common goal, perform specific roles, and depend on each other to achieve the common goal; team success depends on the cooperation of all members. Teams are typically needed for tasks that involve more work, a variety of skills, or some other capability than one person could supply alone.

2.1.2 Building teams

Teambuilding is a way to establish all the right conditions, team behaviors, and management actions for a team to jell and be successful. A *jelled team* is "a group of people so strongly knit together that the whole is greater than the sum of the parts. The production of such a team is greater than that of the same people working in unjelled form. Just as important, the enjoyment that the people derive from their work is greater than what you'd expect given the nature of the work itself" [DeMarco 1987]. Although unjelled teams can successfully achieve their goals, they will be less productive in their work and may produce lower-quality outcomes than jelled teams.

Teambuilding is not difficult, but it is a relatively sophisticated process, with four prerequisites.

1. Every team member must be present during the teambuilding process.
2. The team is given a challenging and important task.
3. The team members understand that success is dependent on the cooperative participation of all members.
4. The team is competently guided and coached.

2.1.3 Building jelled teams

In order for a group to become a jelled team, the following conditions must be fulfilled.

- The team members must share a common goal.
- The group must have cohesion, which has at least two distinct elements [Zaccaro 1991].
  - *Task cohesion* exists when all members are motivated to coordinate their efforts towards achieving the common goal.
  - *Social cohesion* refers to the quality and extent of interpersonal emotional bonds that exists between and among members. Just as a work group cannot successfully achieve its goal without task cohesiveness, it cannot jell without some degree of social cohesion.
- The members must cooperate with each other in pursuit of the common goal.
- The team members must realize that their success depends on the effective performance of every individual in the group.
- The group must communicate freely and regularly.
- Each member must have a defined role that is understood and respected by the other members.
- The boundaries of team membership must be clearly defined; there should be no question as to who is in the group and who is not.

2.1.4 Building successful teams

When a team jells, it improves its chances of success. However, successful teams also need

- a mission that the team believes to be compelling and achievable
- commitments from all members to do whatever it takes to accomplish the mission
- suitable training and other resources necessary for accomplishing the mission
- proper guidance and support

2.1.5 Building high-performing teams

As teams begin to work together, the group members must evolve the processes, work patterns, and interpersonal interactions that will allow them to successfully complete their tasks and achieve their goals. Studies have shown that over time, the activities and behaviors that teams

exhibit typically pass through a pattern of development that can be delineated into four stages: forming, storming, norming, and performing [Tuckman 1965], as illustrated in Figure 2 below.

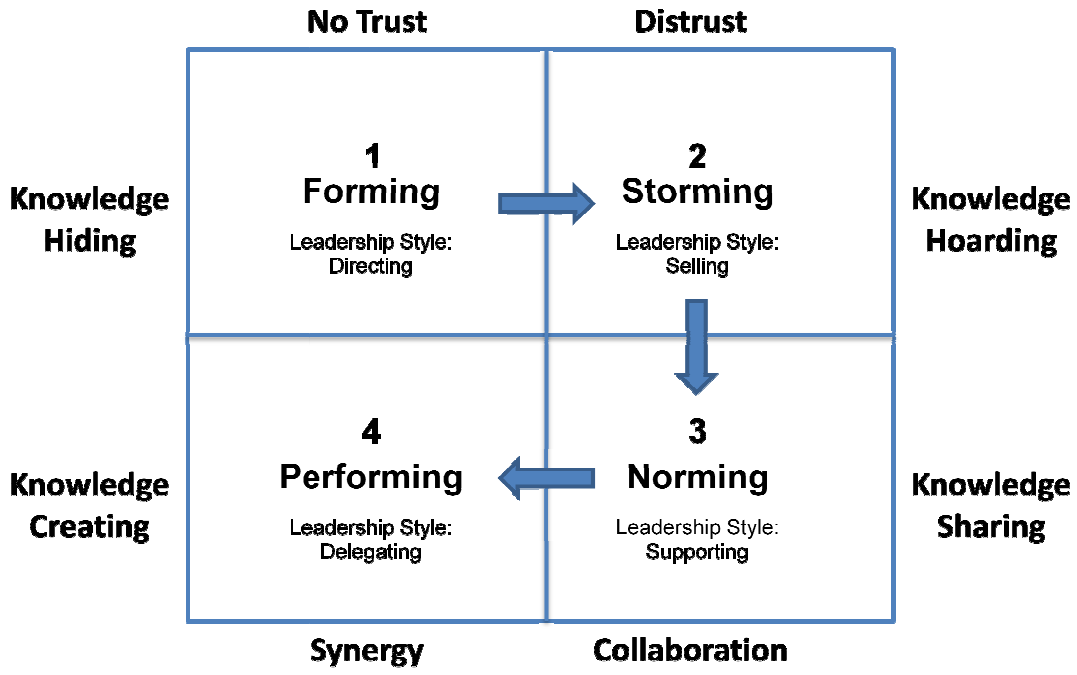

Figure 2: Group Formation Stages

1. The *forming stage* occurs as the team comes together as a working group. The team members learn about on the expectations that they will be expected to fulfill, and focus on getting to know the other members of the group. Although the team members are highly dependent on the leader for guidance and direction, they behave independently of each other. Individual roles and responsibilities are unclear, and team members do not yet know each other well enough to trust each other.
2. The *storming stage* is often emotional, as team members begin to vie for position among themselves and may even challenge the team leader. The group may disagree about the problems they are supposed to solve and the processes that they should use to solve them. Individuals may show resistance to the group influence in order to carve out a niche for themselves. Groups may become stuck in this stage if the members feel that they are being judged or threatened and stop sharing their opinions and views. The team leader, facilitator, and other team members should encourage open and honest communication, while ensuring that the interactions remain constructive.
3. In the *norming stage*, team members reach consensus on rules, behavior, methods, tools, processes, and other standards for working and interacting with each other. Trust grows as individuals get to know each other better, and people begin to feel safe enough to offer honest opinions. Motivation increases as the team gains a clearer understanding of the project and develops some measure of task and/or social cohesion.
4. During the *performing stage*, the team members function as a unit in which group energy is channeled into finding solutions to problems and performing tasks. Members depend on each other and can make decisions without external supervision. Trust has been established,

enabling the free expression of personal opinions; dissent is expected and encouraged as long as it is expressed in accordance with the team's standards. The team members have distinct roles, perform according to agreed-upon work strategies, and follow established plans to reach the goals.

When team members, leaders, and coaches understand the phases of team development, they can work together to keep the group on track as the team proceeds through forming and storming towards effective interactions in the norming and performing phases. Teams should also realize that it is to be expected and entirely normal for the team to return to an earlier developmental stage if the composition of the team changes, the team's performance begins to deteriorate, or problematic team interactions develop.

Knowledge Area 2.2: Team Types, Styles, and Dynamics

This knowledge area describes some of the various models of team types, team styles, and team dynamics which provide a foundational understanding of team work in the TSP.

2.2.1 Transactional and transformational teams

Many organizational psychology models classify groups according to the relationships between the group members and their leaders. There are a number of management styles that fall within a spectrum between transactional leadership at one extreme to transformational leadership at the other [Burns 1978]. TSP teams are modeled on the transformational leadership style.

In the *transactional* leadership model, the relationship is a transaction negotiated between a team leader and an individual team member. The relationship is based on a mutual exchange of valued commodities: the team leader provides rewards or payments in exchange for the team member's satisfactory performance of a specified task. Although team members are expected to work together to produce a given outcome, the transactional team leader tends to view team members as interchangeable parts that can be replaced at any time.

- The team leader's objective is to realize power by exerting control of the team; the team member's objective is to perform the tasks required in order to realize the promised payment.
- All parties are extrinsically motivated; that is, the primary driver for the individual members and the primary reason for the cohesion among the team is the exchange of valued things. Inter-team cooperation is dependent on individuals' perceptions of their fellow team members. The interaction among team members will remain cooperative as long as the members perceive that their individual success depends on input from others; interaction may turn competitive if individual team members perceive others as potential obstacles or threats to their ability to receive the desired rewards.
- The relationship among parties lasts only as long as the agreed-to exchange continues. There is no mutual goal other than the achievement of the individuals' desire for the valued commodities.
- The ends justify the means: the team members place less value on the processes used or the quality achieved in attaining the outcome and more value on reaching the goal.

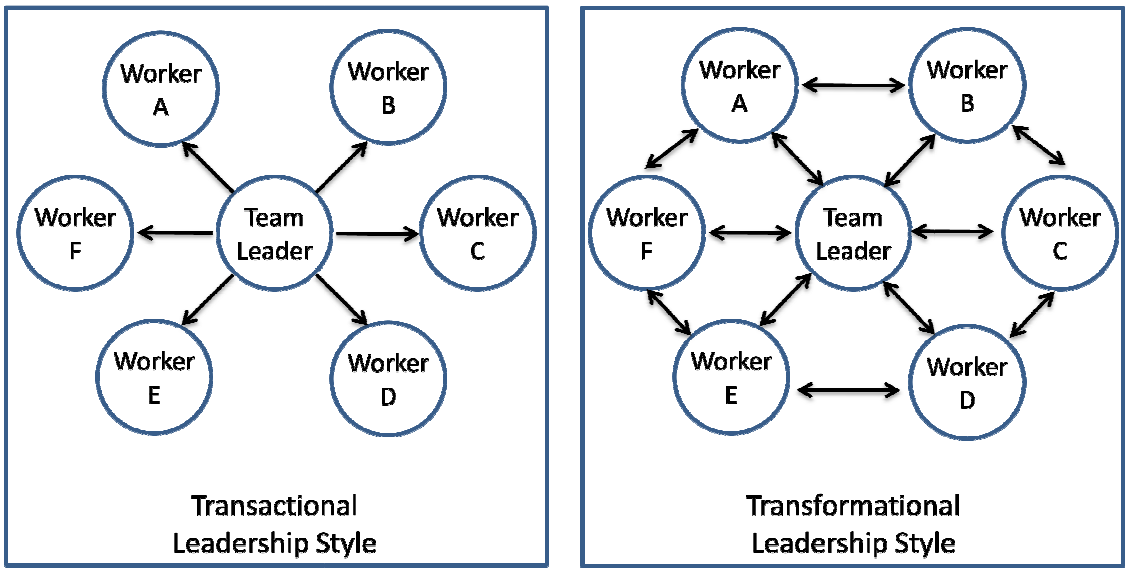

Figure 3: Transactional and Transformational Leadership Styles

With *transformational* leadership, the team leader encourages and empowers team members to work collaboratively to achieve an objective.

- Both the team leader and team members have a common objective: to find new ways of looking at old problems and then create appropriate solutions. Although financial and other rewards are important benefits, the real motivation driving transformational teams is achievement.
- The team members are intrinsically motivated to work on the team; the work and the leader's vision inspire the team to strive both individually and collectively to reach the desired outcomes and achieve the satisfaction of having succeeded.
- The team leader views the team as a finely-tuned machine in which each part has a specific function. Members are valued for their skills and experiences, and are encouraged to continue learning and developing while working on the team.
- The means are as important as the end: the team leader and team members value the process as much as they value the achievement of the goal.

2.2.2 Self-directed teams

Self-directed teams work together to perform a function, deliver a service, or produce a product. They also take on the management of that work and perform functions that are traditionally the responsibility of supervisors or managers. This enables the team's managers to teach, coach, develop, and facilitate, rather than simply to direct and control [Williams 1995]. Self-directed teams are best suited for performing creative knowledge work. The major characteristics of self-directed teams are as follows.

- The members of a self-directed team have a feeling of membership and belonging, are committed to common team goals, and share a common dedication to excellence.
- Self-directed teams negotiate and manage their own commitments. They adopt their own processes, set their own quality standards, and identify any problems that might impede the team's ability to meet their commitments. They will participate in team actions to resolve

the problems, or, as necessary, will escalate problems to whatever management level is required to resolve them.

- Members of self-directed teams behave honestly and respectfully towards each other, and engage in frequent and honest communication with each other, their management, and their stakeholders.
- The members of self-directed teams behave responsibly. That is, they act as if they personally own the project and have a deep personal commitment in the team's success.

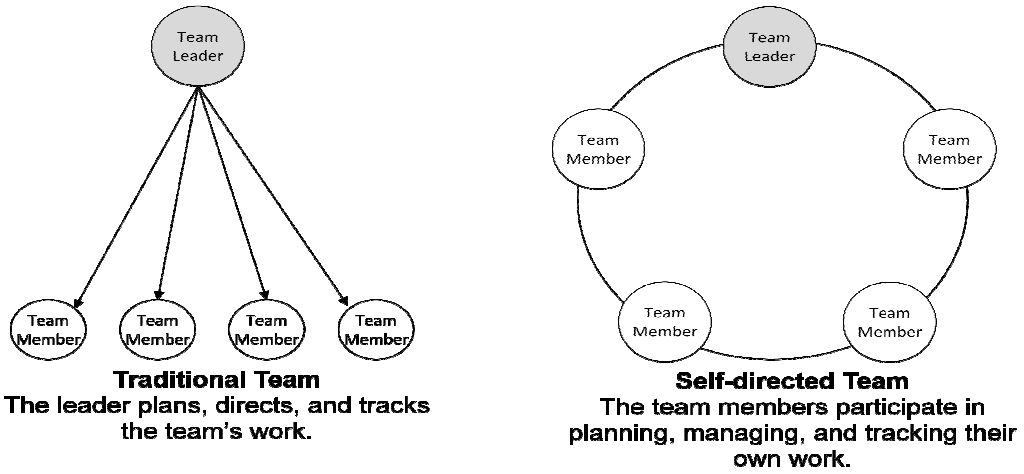

Figure 4: Traditional and Self-Directed Teams

2.2.3 Team types

Teams may be categorized into three types according to the way they function under changing conditions [Berne 1966]: the work group, the process group, and the combat group.

- The term *work group* describes how a team functions under "normal" circumstances; that is, there is no external or internal stress to detract the members from going about their usual jobs. The work group model is the ideal state to maximize team efficiency and effectiveness.
- When a team experiences problems with its own internal working relationships, it diverts its attention from getting the job done and focuses much of its energy to solving the team's working problems. Because the team members are most concerned with their internal mode of operation, the team has become a *process group*, in which the focus is on the group process instead of the group mission. This behavior often results when team members become uncomfortable or unclear about their roles or assignments.
- The *combat group* is preoccupied with real or perceived external threats and focuses its attention on dealing with those threats instead of on doing its work. To get back on track, the team should address and resolve real problems, or spend time to identify the perceived threat and devise an action plan to resolve the issue.

Team members, leaders, and coaches should understand these group types so that they can recognize when performance problems arise and help the team to take the appropriate steps needed to return to work group conditions.

2.2.4 Team working styles

Teams differ not only in the way by which they are led or how they function under pressure, but also in the mechanisms by which they coordinate and control their efforts when working to achieve a common outcome. Larry Constantine [Constantine 1993] developed a reference framework to define the range of paradigms describing how teams function when conducting project tasks. The model matrix has four reference paradigms or work styles that correspond to stereotypical extremes of group control and coordination, as shown in Figure 5. These are the closed, random, open, and synchronous group styles.

- The *closed group* style is based on the traditional hierarchy of authority, managed from the top down. The closed group style is best applied when conducting routine projects in which there is a specific job to do and a clearly defined way to do it. The closed group style is beneficial when resources are limited and the job must be done as quickly and efficiently as possible. The weakness of this style is that it doesn't fully utilize team-member skills or foster innovation.
- The *random group* style relies on someone to take the initiative to coordinate the team's activities. Random groups strive for a free flow of ideas in order to spark creativity and allow everyone to be heard. The random group style is best applied when a creative solution is needed for a common problem. The benefit of the random style is that it fosters innovation and encourages team members to think "outside the box." The weakness of the style is that it is time-consuming and recourse-intensive.

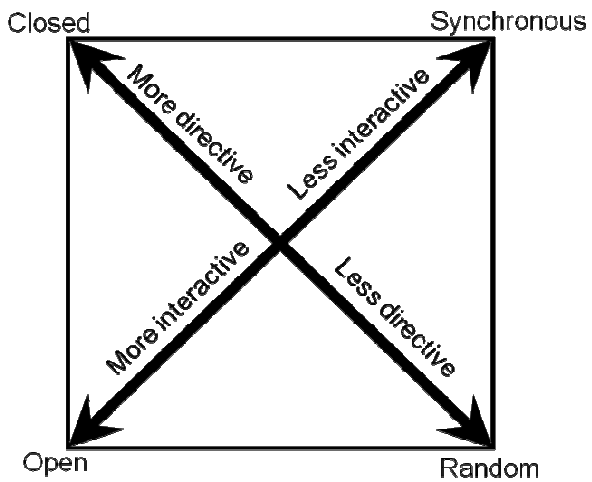

Figure 5: Team Working Styles

- The *open group* style is a process-oriented collaboration in which team members essentially share the task of managing themselves. Open groups typically have high energy and team spirit, and members adjust their behavior to support or assist each other. Decisions are negotiated and adopted by mutual consent. The open style is best applied to complex problem-solving tasks, including software development work. The benefit of open style groups is that they are able to share information to make plans and develop appropriate strategies. The weakness of the open style is that the team can become bogged down in endless debates.

- The *synchronous group* style is characterized by individuals working independently in parallel. The members have a high degree of personal commitment and a shared vision for performance outcomes. The synchronous style is best applied to situations in which highly-skilled individuals are performing repetitive yet critical tasks that are related but not strongly interdependent. The benefits of the synchronous style are realized in the efficiency and smooth performance of team operations. The weaknesses of the synchronous style are that group cohesion may suffer due to a lack of overt communication, and the team generally does not respond well to change.

Since no single style is best for all types of group activities, groups are most productive and creative when they understand the paradigms and use them to adjust their responses to the immediate situation.

Knowledge Area 2.3: Team Formation and Membership

This knowledge area describes important parameters that should be considered when forming and populating teams.

2.3.1 Team size

The team is the basic unit for TSP work. Its members must meet all of the team membership criteria listed below in section 2.3.3, and the team size generally should be less than 15 members. Teams of 8 to 12 members are considered the ideal size. Teams with more than 15 members should consider dividing the work along architectural lines and using two teams working under the TSP multi-team structure. The TSP multi-team structure is the most appropriate approach for projects that are too large for a single team.

2.3.2 Team scope

When forming a TSP team, it is important to include all appropriate members; it is equally important to limit the team scope to those who will be actively involved and fully committed contributors. Team members must all share common goals and be willing to make a significant commitment to meeting them. This implies that team members should spend a substantial portion (or all) of their time performing the project tasks necessary for attaining team goals and accomplishing the team mission. Part-time team membership should be limited.

2.3.3 Team membership

When selecting the team members for a new team or to add to an existing team, the following criteria should be considered.

- **Skills** – Team members should possess most or all skills needed for the job before they are selected for team membership. Although additional training can be provided after members have joined a team, training takes time; therefore, whenever possible, it is preferable to choose people who already possess the critical skills needed. A counter-consideration is that untrained candidates may be available only because they lack the training that would have resulted in their membership on other teams. In these cases, it may be better to hire and then train the untrained candidate team members.

- **Aptitudes** – While skills are trainable, aptitudes are not. Therefore, it is essential to examine all potential candidate team members to identify their aptitudes, and then to choose those individuals who possess the aptitudes that best match the tasks that team members will perform.
- **Interests** – Candidate team members' interests often provide the best indicators of their skills and aptitudes. Individuals who clearly are not interested in doing the work the team has to do – regardless of their skills and aptitudes – are not good choices for team membership.
- **Teamworking ability** – The extent to which a candidate team member is likely to be a "team player" is the most difficult criterion to identify, but it is also the most important. When screening for this criterion, multiple interviewers can be helpful. If the candidate comes from inside the organization, the interviewers are likely to know the candidates or some of their prior teammates, and will therefore be able to provide valuable insight about the candidate team members' potential ability to fit in well with the rest of the team.

2.3.4 Team member TSP training

Team members must be trained in TSP before serving as members of a TSP team. There are two basic methods for receiving the required TSP training.

- **Full training** – Before the team begins its work and has its initial team launch, all team members must complete all of the PSP or personal process training (*PSP for Engineers, Parts 1 and 2*, or both *PSP Fundamentals* and *PSP Advanced*, or the analogous training suite for non-software developers). The team members should also receive basic training on how to use whatever TSP support tool used by the team. This training can be followed with optional advanced TSP courses, as needed or requested.
- **Just-in-time training** – Prior to the team's initial launch, all members must complete basic PSP training (*PSP for Engineers, Part 1* or *PSP Fundamentals*) or basic personal process training, and they should receive instruction on how to use the chosen TSP support tool. Following the launch, each team member <u>must</u> complete the appropriate follow-on PSP course (*PSP for Engineers, Part 2* or *PSP Advanced*) and any additional training deemed necessary for effective team working.

After the team members have received the required PSP training and have mastered basic TSP topics, additional training in role member skills may prove to be beneficial. Team members who are interested in advancing their knowledge of process improvement may wish to pursue additional training on teaching PSP and TSP, becoming a TSP coach, or team leadership and management topics. Team members who do software or systems engineering work may wish to pursue certification, such as the SEI-Certified PSP Developer credential. The benefits of becoming certified in a particular discipline not only provides individuals with a valuable professional capability, but also provides managers and fellow team members with the assurance that they have the knowledge and qualifications needed for effective performance as a TSP team member.

Knowledge Area 2.4: Team Member Responsibilities

This knowledge area addresses the team's responsibilities for helping teams to be fully effective. These responsibilities are as follows.

- Communicating
- Meeting commitments
- Accepting and performing a team role
- Participating fully in team activities
- Following the team's defined processes
- Being prepared
- Acting rationally

2.4.1 Communicating

Communication is the single most important element in both building and maintaining teams. To have a fully effective team, all team members must communicate freely and openly with each other and with the team leader, team coach, and higher management. Good team communication allows the team to quickly resolve understandings and disagreements, identify and address problems, and maintain the rapport and trusting environment required for effective teamwork.

Team communication can be facilitated by locating the team members in a common workspace and holding frequent team meetings. When exchanging information, team members should share any and all relevant information, even seemingly insignificant facts or data. Many teams have failed because one member was aware of a potential problem that could have been easily solved early on, but the information was not shared because it didn't seem important at the time or because the team member assumed that everyone else had the same information.

2.4.2 Meeting commitments

In simplest terms, a *commitment* is a promise by a person, a group, or an organization to do something. By definition, a commitment includes a delineation of the task(s) that will be performed and a timeframe or end date for completing the task. To make a responsible team commitment, all team members must participate in making their plan. They must

- find out what management wants
- strive to make a plan that meets management's needs
- work as a team to produce a plan for the entire job
- reach team consensus on the resulting plan and agree to the commitment
- participate in negotiating the plan and commitment with management

The success or failure of most group activities often rests upon the ability of every member of the group to consistently meet their commitments. If team members find that they will not be able to meet a commitment as planned, they should promptly inform the rest of the team and assist in determining and carrying out the actions needed to resolve or mitigate the resulting issues.

2.4.3 Accepting and performing a team role

TSP teams have eight designated member roles (see Knowledge Area 2.5), and every team member is expected to accept and perform the duties of one or more of these roles.

- When the team has more members than roles, members without designated roles should be alternate role managers and support the primary role managers in their duties.
- When the team has more roles than members, some members must handle two or more roles.

2.4.4 Participating fully in team activities

In addition to accepting and performing a team role, team members must attend all team meetings, assist other members when they need help, actively contribute to the team launches and relaunches, and act in ways that enable the success of the team and all of its members.

2.4.5 Following the team's defined processes

As part of a TSP team, all members are expected to work in accordance with the team's defined process. They must make and track personal plans; record their time, size, and defect data; do quality work; and strive to meet all personal and team commitments.

2.4.6 Being prepared

Preparation is the key to success for most technical and development activities. Before joining a team, potential members must ensure that they have the requisite knowledge and skills or make a firm commitment to acquire needed training as soon as possible.

2.4.7 Acting rationally

To be fully effective on a team, all members must agree to use facts and data, rather than feelings or emotions, to resolve team issues.

Knowledge Area 2.5: Team Member Roles

This knowledge area discusses the types of roles on TSP teams and delineates the responsibilities required of the various team member roles.

2.5.1 TSP team roles

A role can be defined as the rights, obligations, and expected behavior patterns associated with a particular social status. The TSP roles describe what each person on the team is expected to do and transform team members from followers to co-managers of the project. The TSP roles cover all aspects of team management from customer interface issues to testing, and from process management to quality control. They also help team members to develop their own plans, manage and track the quality of their work, and decide which processes to use in carrying out the project. There are two types of team member roles in the TSP: the general team member role, and eight (or more) designated manager roles. Each member of a TSP team is expected to fulfill the general team member role responsibilities in addition to the duties of one or more manager roles.

2.5.2 The TSP general team member role

All team members are expected to follow a disciplined personal process to plan, manage, and report on the tasks done to carry out the team's work, and they must meet their personal commitment to produce quality products. They are also expected to keep the team leader and team members informed of their project status.

In order to produce quality products, team members should prevent defects whenever possible, remove defects as early in the process as possible, and record data on every defect that they find, and then use the data to improve their personal process.

Team members are expected to cooperate with the other members of the team to maintain an effective and productive working environment. Effective teamwork requires all team members to

- work objectively to settle team issues or resolve project problems in the best interest of the team and its goals
- call on other team members for help whenever needed
- support other team members when they need help
- take responsibility for one or more team roles
- participate in team activities and make sure that their views and ideas are known and understood
- listen to other team members and strive to understand their views and ideas

2.5.3 The TSP manager roles

The standard TSP roles divide the principal team management responsibilities among all of the members. Roles allow each member of the team to handle a specific subset of the overall job so that the necessary key tasks for running a team are handled expeditiously and effectively. Assignment of roles also enable the team to better mitigate risk, since each role manager is expected to anticipate problems and, if they arise, to address them personally, to alert the team member with the appropriate role responsibility, or to work with senior management or the customer to resolve the issue. The shared responsibility for team management tasks among all of the team members also has the benefit of accelerating teambuilding and enabling the team members to focus more effectively on performing the actions necessary to accomplish the team's assigned mission.

To ensure maximum efficiency in carrying out the role duties, team members should be familiar with the expectations not only of their own personal role, but also those of all the other roles; this helps to avoid duplicated effort. In addition to helping to ensure that the normal issues of running a team are handled expeditiously and effectively, the assignment of TSP roles also accelerates teambuilding.

There are eight standard TSP manager roles; each is described in more detail in the remaining sections of this knowledge area.

- Planning manager
- Process manager
- Quality manager
- Support manager
- Customer interface manager
- Design manager
- Implementation manager
- Test manager

2.5.4 The planning manager role

The planning manager's role is to help the team run a well-planned and tracked project. The planning manager is responsible for ensuring that

- the team is always working to a defined and documented plan
- plans are generated in the launch for the team as a whole and for each individual team member, in which all known tasks are listed and estimated
- the team always has a balanced plan in which the workload is distributed as evenly as possible and has all team members completing their work at roughly the same time
- plans are revised at every relaunch, whenever the project schedule or resources change substantially, or when it is clear that the team's plan has become unbalanced
- the schedule, resource, size, and productivity sections of the project report are produced for the phase or cycle, and project postmortems

After the launch or relaunch, the planning manager tracks team progress against the plan and reports to the team on project status. Each week, the planning manager

- ensures that all team members update their personal task and schedule plans
- uses individual plan data to update the team's task and schedule plans
- analyzes the updated plan data to keep the team and management informed of likely phase or cycle, and project completion dates
- alerts the team leader when issues arise or risks to the plan become apparent
- supports the team leader in producing weekly management and customer status reports

2.5.5 The process manager role

The process manager's role is to ensure that the team has defined processes available for major development, management, and team functional activities. Among the process manager's specific duties are

- ensuring that the team always follows a defined and documented process
- leading the team in defining or developing the processes that the team needs and then ensuring that the processes are used to guide the team's work
- assisting the team in identifying areas in which team members are having problems in following the defined process and ensuring that process problems are quickly resolved
- ensuring that all team members report their process data in a timely way
- reporting weekly to the team on the status of all team process development and analysis work
- alerting the team and team leader when process problems need their attention
- making sure that the project notebook is complete and up to date
- producing the process section of the project report during the phase or cycle, and project postmortems
- gathering and analyzing process improvement proposals (PIPs), determining any process changes required to instantiate the PIPs, making recommendations on how or whether to make the changes, and overseeing the implementation of PIPs that the team decides to enact

2.5.6 The quality manager role

The quality manager's role is to lead the team in producing and following the quality parameters for the project, to provide timely analysis and warning of quality problems, and to perform effectively as the team's inspection moderator. Among the quality managers duties are

- leading the team in developing and following the quality plan
- tracking product and process quality measures regularly, alerting the team and management whenever quality problems require special attention, and recommending corrective actions
- maintaining a focus on product and process quality throughout the project
- alerting the team whenever the defined process is not being followed and recommending corrections for the problem
- ensuring that the members gather their quality data regularly and reporting weekly to the team on quality measures and product quality status
- analyzing team quality data and ensuring that these analyses are available for team reference
- updating the quality summary for the system and for each of its parts
- ensuring that a qualified moderator is available to lead team inspections or acts as inspection moderator
- maintaining the data to produce the defect, yield, ratio, rate, and component sections of the project report for the phase or cycle, and project postmortems

2.5.7 The support manager role

The support manager's primary responsibility is to ensure that the team has the proper tools and methods needed to do the project work. The support manager also handles the team's configuration management and change control functions, and acts as the team's reuse advocate. Among the support manager's duties are

- ensuring that the team has an appropriate development support system and tracking the performance and effectiveness of that system
- leading the team in developing or obtaining special support tools or facilities
- ensuring that the team members are familiar with support tools and, where necessary, trained in their use
- managing the team's configuration management system and maintaining the master copies of all controlled items and versions
- leading the configuration control board
- maintaining a list of potentially reusable parts and alerting the team to reuse opportunities
- tracking and reporting weekly to the team on the status of all support procurement and development work and on reuse status and opportunities

2.5.8 The customer interface manager role

The customer interface manager's role is to understand the customer's wants and needs and then to lead the team in providing a product that delights the customer. Accordingly, the customer interface manager coordinates the team's interactions with the customer by

- maintaining a focus on the customer's needs throughout the project
- leading the team in producing, refining, and verifying the product requirements
- establishing team standards and procedures for documenting and reviewing the product requirements

- ensuring that the customer agrees with the product requirements
- ensuring that all requirements assumptions are identified, documented, and verified
- defining prototypes (if needed) to help the customer understand proposed product features
- working with the customer to establish acceptance test criteria and plans
- documenting agreements with the customer, and ensuring that the customer reviews and approves the documents
- managing the requirements change process and coordinating changes with the configuration control board
- tracking and reporting weekly to the team on the status of the requirements standards and requirements development

2.5.9 The design manager role

The design manager's role is to lead the team in producing a superior design. Among the design manager's specific responsibilities are

- leading the team in producing, refining, and verifying the product design
- establishing the standards and procedures that the team will use to produce the design materials
- ensuring that the design and its documentation are of high quality
- maintaining a focus on design issues throughout the project
- identifying and resolving all design issues, and documenting and confirming the resolutions
- managing the design change process and coordinating changes with the configuration control board
- reporting weekly to the team on the status of design standards and product work

2.5.10 The implementation manager role

The implementation manager's role is to produce an implemented product that is of high quality. Among the implementation manager's specific responsibilities are

- ensuring that the implementation fully conforms to the design
- establishing the standards and procedures the team will use to produce the product implementation and its documentation
- ensuring that the team has standards for coding, size counting, language, and documentation
- identifying and resolving all implementation issues, and documenting and confirming the resolutions
- leading the team in producing, refining, and verifying the product implementation
- leading the team in measuring and identifying any performance and size issues
- leading the team in planning for and handling product packaging, distribution, and installation problems
- maintaining a focus on implementation issues throughout the project
- managing the implementation change process and coordinates changes with the configuration control board
- reporting to the team weekly on the status of implementation standards and product implementation

2.5.11 The test manager role

The test manager's role is to ensure that the system is thoroughly tested and performs all important functions properly. Among the test manager's responsibilities are

- leading the team in developing comprehensive test plans
- supporting the customer interface manager to get acceptance test criteria defined and agreed to by the customer during the requirements phase
- defining and planning the system tests during the design phase
- defining and planning the integration tests during the implementation phase
- supports the team members with planning and executing all test activities
- analyzing data from every test phase to identify defect-prone product elements and working with the quality manager to identify areas that need reinspection or retest
- maintaining a defect density map of all product components and the overall system for every test phase
- leading the team in maintaining a focus on testing throughout the entire development process
- tracking and reporting weekly to the team on the status of the team's test planning, development, and execution work

2.5.12 Other management roles

The eight standard TSP roles cover most normal team activities. However, additional roles may be needed when a project involves work that is not typically covered by the standard roles or when special emphasis is needed for some areas. Because team members need time to work on their project development tasks and role assignments can take away from project development time, it is important to ensure that any new role assignments are truly needed and that team members are willing and able to handle them. Therefore, whenever possible, additional responsibilities should be added to existing roles if the new work can be combined with or added to similar related role tasks. If additional roles are created, the selected role managers should immediately define and document their role responsibilities, then review the role definitions with the entire team and get their agreement. Some of the additional roles that teams might consider include (but are not limited) the following.

- Dependency manager
- Hardware interface manager
- Installation manager
- Performance manager
- Privacy manager
- Reuse manager
- Safety manager
- Security manager
- Subcontract manager

Knowledge Area 2.6: Team Leader Role

The TSP team has the overall responsibility to higher management for the team's success. Therefore, the leader must provide the enthusiasm, energy, and drive that will convince the team members to join the leader in achieving a set of common goals. To accomplish this end, the TSP

team leader must carefully balance several roles and numerous responsibilities to both the TSP team members and the organization's management. This knowledge area lists the roles that a TSP team leader must fulfill and discusses some of the tasks and responsibilities that accompany the various roles.

2.6.1 The TSP team leader's various roles

The TSP team leader must simultaneously fulfill several different types of roles, each of which has its own unique blend of responsibilities and priorities (which are described in the ensuing sections of this Knowledge Area). The TSP team leader must function as

- the designated manager of the team (the team leader's primary role)
- management's representative to the team
- the team's representative to management
- a team participant
- supervisor to some (or all) of the team members

A successful TSP team leader has achieved balance and synergy among these five potentially conflicting roles. Humphrey [Humphrey 2006b] offers the following guidelines for balancing the team leader's various functions. (Other team leader guidelines are provided in Appendix D.)

- Use logic, data, and persuasion to guide the team members in making decisions and as the foundation for reports of team progress or team problems to higher management.
- View assertions of authority as a personal failure.
- View leadership and motivational actions as personal achievements.
- Remember to honor and respect all team members for their views, opinions, and individuality.
- Guide the team in making decisions that the team leader can comfortably support and defend to management.

2.6.2 The TSP team leader's role as team manager

The team manager function is the largest part of the TSP leader's job, and not surprisingly, is also the single most important factor affecting the team's performance during a TSP project. Therefore, the team leader fulfills two primary duties that contribute to the TSP team's and the project's success: team motivation and project resource management.

As a motivator, the team leader must establish and maintain an attitude that conveys personal commitment to the project and its goals, and demonstrate trust that the team members share this commitment and will do the best job possible to fulfill the commitment. The most effective way of motivating teams to act responsibly in meeting their commitments is to lead by example.

- Adhere to the agreed team standards and processes
- Follow disciplined personal practices
- Record process data regularly and use the data to improve personal practices
- Emphasize quality and strive to produce high quality products
- Respect data confidentiality
- Protect the team from diversions and time-consuming distractions
- Enable and maintain open and honest communication

The team leader may also perform traditional business oversight duties, such as

- managing the team's budget and other resources
- managing the project's staffing and training requirements
- providing a suitable and conducive environment for teamworking
- ensuring that the project stays on schedule
- ensuring that all team members are working productively and effectively
- supervising individual team members

2.6.3 The TSP team leader's role as management's representative to the team

As the representative of management to the team, the team leader must

- convey management's needs to the team and clarify requirements as needed
- provide the team with interim guidance if management needs are unclear, and then check with management to provide clarification
- maintain a consistent focus on meeting management's goals
- relay important management communications to the team

2.6.4 The TSP team leader's role as the team's representative to management

As the team's representative to management, the team leader must

- work with the coach to brief all participating managers on their roles in the launch process, and ensure that the managers provide appropriate information to the team during launch meeting 1
- provide any needed insight and understanding to support the team's actions and decisions
- identify any team assumptions or decisions that may be viewed by management as contentious and review these contentious items with management for input
- keep management apprised of project status, progress, and key issues
- mediate negotiations to obtain needed changes in resources, schedule, budget, requirements, or major parameters of the project

2.6.5 The TSP team leader's role as an implementation team member

As a member of the implementation team, the team leader must

- participate as a peer, rather than as a leader, in team discussions
- give others a chance to speak and express their views before stating an opinion
- make the effort to understand the logic for counter positions when the team members have differing positions or opinions
- be willing to accept the team's choice when any of several alternative approaches are of equal merit
- revert to the team leader role and require the team members to defend their choice if the team leader feels that the team has made a wrong choice
- work with the team to achieve a mutually agreeable decision whenever possible

2.6.6 The TSP team leader's role as supervisor

As a member of the implementation team, the team leader should

- avoid intimidating or inhibiting team discussions or decisions by using (or even alluding to) the fact that the team leader has supervisory authority over some or all of the team members

- provide private guidance as appropriate if team members under the team leader's supervisory authority have performance issues that need to be addressed and improved
- refrain from using team member data or estimates when conducting performance evaluations
- consider TSP factors (such as team member behavior, commitment, attitude, quality management, and willingness to contribute to team effectiveness) when conducting performance evaluations of team members under the team leader's supervision

Knowledge Area 2.7: Coach Role

This knowledge area describes the general roles and responsibilities of the TSP coach. Guidelines for handling specific TSP coaching situations can be found in Appendix C.

2.7.1 TSP coach objectives

The TSP coach's main objective is to provide the required skill, discipline, insight, and outside perspective that teams and individuals require to be successful. The coach uses tailored approaches within a firm set of principles to meet the needs of the team and its members. The coach also works to motivate and support the team in meeting its established goals.

2.7.2 TSP coach roles

To achieve their main objective, TSP coaches have five overarching roles to perform: TSP expert and advocate, individual team member coach, team coach, team leader coach, and process evaluator. Each of these roles is discussed in greater detail in sections 2.7.3 through 2.7.7.

2.7.3 The TSP coach's role as TSP expert and advocate

As a TSP expert and advocate, the coach should perform the following activities.

- Guide the team during the launch in following the TSP process, entering and using data, and using the available TSP support tool.
- Provide any needed explanations if there are questions about the logic for the TSP process or its elements.
- Refrain from selling the TSP and let the team members learn to appreciate its benefits through using it.
- Attempt to convince team members who disagree with using all or part of the TSP to follow the process, enlisting the team's, team leader's, or management's help in doing so.
- Respect and, to the extent possible, enlist the help of TSP skeptics.
- Be willing to explain process activities and to guide the team in performing them, but insist that the team does its own work.

2.7.4 The TSP coach's role in coaching individual team members

As the coach for each individual team member, the coach should perform the following activities.

- Ensure that all team members participate fully in the launch and ongoing team activities.
- Ensure that all team members are given the chance to contribute.
- Observe how each of the team members work and identify areas for improvement.
- Allow the team members to make their own mistakes, but make sure that they do not do anything that is destructive to themselves, the team, or the team's mission.

- Offer private help to team members who make mistakes and enable them to learn from their mistakes.
- Guide the team members in accepting challenges that will stretch their skills, but help them to avoid commitments that are clearly beyond their capabilities.
- Keep the team members focused on using TSP processes to improve their personal performance.

2.7.5 The TSP coach's role in coaching the team

When acting as coach to the team as a whole, the coach helps the team members to become a cohesive working unit by doing these things.

- Let the team make its own mistakes, but ensuring that it does not do anything that would be destructive to itself or its mission.
- Guide the team in conducting self-assessments of its performance as a cohesive group and in devising ways to improve.
- Provide objective assessments of the team's performance as a cohesive group and offering suggestions for improvement.
- Choose an appropriate time to point out instances when the team did not do its job in the best or most effective way and offering suggestions as to how the team could have acted more effectively.
- Keep the team focused on using the TSP process to improve its performance.

2.7.6 The TSP coach's role in coaching the team leader

When coaching the team leader, the coach should do the following.

- Observe the team leader's performance as a leader and identifying areas for improvement.
- Let the team leader make mistakes, but ensuring that these mistakes are not destructive to themselves, the team, or the team's mission.
- Regularly assessing the team leader's effectiveness in helping the team to become a cohesive and effective working group.
- Choose an appropriate time to point out instances when the team leader did not perform as a leader in the best or most effective way and offering suggestions as to how the leader could have performed more effectively.
- Keep the team leader focused on using TSP processes to improve the team's performance.

2.7.7 The TSP coach's role as process evaluator

As a process evaluator, the TSP coach should do the following.

- Observe how well the process supports and guides the team in its work.
- Note areas for possible process improvement and discuss the observations with the team during the launch postmortem, the phase or cycle postmortem, personal team member postmortems, or checkpoint reviews.
- Ensure that improvement suggestions are recorded in PIPs and submitted to the process manager for evaluation and implementation

2.7.8 The TSP coach's general responsibilities

The coach is responsible to the team leader for conducting the launch and the follow-on team coaching. The coach is also responsible to higher-level management for ensuring that once the

team is launched, it is capable, willing, and motivated to do the job with which it has been charged.

2.7.9 The TSP coach's launch preparation responsibilities

Before conducting a TSP team launch, the TSP coach has the responsibility to ensure that several preliminary conditions have been met. The coach must carry out these responsibilities.

- Ensure that all of the team members have the training required to do the tasks required in the launch and for their subsequent work.
- Support the team leader in launching and coaching the team.
- Monitor and assist the individual team members in improving their personal performance and enhancing their ability to work as a coordinated and cooperative team.
- Obtain management participation in the launch process.
- Ensure that the team leader understands the required management participation and knows how to advise the participating managers on what they are to do during the launch.
- Ensure that the proper meeting facilities, electronic support services, and all other logistical needs are available when needed.

The coach is also responsible to management for ensuring that the team is properly led.

2.7.10 The TSP coach's responsibility to the organization's management

The coach's principal responsibility is to the management of the organization that is sponsoring the TSP effort. To fulfill this responsibility, the coach must guide management in enacting all of the necessary steps to successfully implement TSP in their organization, and also must work with the team to ensure its success. The coach must assure management that the team leader and team members are performing their TSP roles in a professional and satisfactory manner, and that they are working with due diligence to meet the goals that management has requested of the team.

2.7.11 The TSP coach's responsibilities to the team leader

The coach's obligation to the team leader is to work with the team leader in enabling the team do its job successfully. When the team has problems, the coach works with the team leader to find an amicable solution. The coach is also responsible for overseeing the team leader's compliance with the TSP process, and for guiding the team leader's efforts in ensuring that the team is also following the TSP process.

2.7.12 The TSP coach's responsibilities to the team

The coach is responsible for overseeing the performance of the team as a working unit and for helping the team members to follow the agreed-to processes for performing its work.

2.7.13 The TSP coach's responsibilities to the individual team members

The coach's responsibility to the individual team members is to help them to use and understand their personal data and to respect the team members' privacy by keeping private information confidential. The coach is also responsible for ensuring that all team members are following the process and adequately performing their assigned role member duties.

2.7.14 The TSP coach's responsibilities to self

TSP coaches have a responsibility to themselves to avoid situations in which they end up with conflicts in their responsibilities to the various constituencies. If such conflicts arise, they should act honestly and ethically, and where necessary, recuse themselves from questionable situations by informing higher management of the conflict and asking for help in resolving the problems.

Competency Area 3: Project Planning with TSP

Project planning with TSP begins when an organization makes the decision to implement the new technology using one or more small pilot projects. The organization works with a certified TSP coach to choose pilot teams and to provide them with training in PSP or process improvement techniques and in TSP project planning and tracking methodologies. Once it is trained, the team plans the project work by participating in a TSP launch. The TSP launch process is actually a series of teambuilding activities that are organized into ten meetings. The TSP coach works with the team's members, leader, and managers to form an understanding of the project requirements, determine team goals, select team member roles, and make plans for producing the product. The team presents its plan to management and after receiving management acceptance for the plan, or an alternative plan, begins to work on the project.

The knowledge areas in this competency area describe the requirements, activities, and guideposts for implementing pilot TSP projects and launching TSP teams, as follows.

3.1 Change Management Fundamentals - This knowledge area describes some of the fundamental concepts of change management and technology introduction that can help to facilitate the introduction of new technologies, processes, and practices into an organization.

3.2 Piloting TSP in an Organization - This knowledge area describes the general requirements and guidelines for initiating and conducting TSP pilot projects.

3.3 Preparing Management and Teams for TSP Implementation - This knowledge area describes the training and pre-launch preparations required to prepare executives, managers, team leaders, and team members for effective participation in TSP implementations.

3.4 The TSP Launch Meetings - This knowledge area provides an overview of the TSP launch and a description of each of the meetings that make up a TSP launch.

3.5 The TSP Relaunch - The team relaunch is nearly the same as the launch except that it is done by a team that has already completed an initial launch of the same project. This knowledge area describes how the relaunch differs from the launch, explains when and how to conduct a relaunch, and delineates the inputs and outputs for the relaunch.

References: The material covered in this competency area is detailed in these primary sources.

[Conner 1982]
[Rogers 2003, Chapters 5 and 7]
[Humphrey 2006a, Chapters 4 through 16]
[Humphrey 2006b, Chapter 7]

Knowledge Area 3.1: Change Management Fundamentals

This knowledge area describes some of the fundamental concepts of change management and technology introduction that can help to facilitate the introduction of new technologies, processes, and practices into an organization.

3.1.1 Technology transition

The process of inserting a new process, practice, methodology, or technology into an organization is often referred to as *technology transition*. The complete process of technology transition includes planning for introduction of changes or innovations, small-scale deployment of the proposed changes or innovations into the organization through one or more pilot projects, large-scale roll-out to multiple projects or divisions, and broad adoption and routine use across the organization.

3.1.2 Technology transition management principles

Executives or managers often attempt to introduce new technologies into their organizations, with varying degrees of success. These introduction initiatives often fail, not because of anything about the proposed change, but because of mistakes made by the individuals responsible for introducing the change. These individuals, called *change agents*, often forget that it is not the organization that must change; only individuals can change (or refuse to change). In order for changes to be willingly accepted, change agents must ensure that the affected personnel are provided with necessary support factors, including the following.

- *Awareness* of what the change involves, why it is being made, how it will affect their work, and the anticipated value to be realized from making the desired change
- A *desire to support and participate* in the change, in the form of both intrinsic rewards (such as an increase in job satisfaction) and extrinsic rewards (such as bonus pay or awards)
- *Knowledge, skills, or training* needed to successfully implement the change or innovation
- *Opportunities* to practice new skills and behaviors
- *Support* from management and *reinforcement* to sustain the change

Change agents should also understand that different people have differing intrinsic reactions to change; some individuals are naturally open to change, whereas others will be more resistant. Therefore, change efforts are more likely to succeed if they are initially aimed at the more change-adept personnel who can later serve as positive role models for their change-resistant peers.

3.1.3 Stages of the technology adoption process

Adoption of a change or technological innovation occurs over time in several stages, at each of which, a decision is made to continue with or reject the adoption. At the individual level, Rogers has identified five stages [Rogers 2003]. These stages are knowledge (awareness of the innovation), persuasion (increased interest), decision to accept or reject, implementation (trial use), and confirmation (adoption). These stages are also seen in the adoption process at the organizational level, as described by the Commitment to Organizational Change model [Conner 1982], shown in Figure 6.

Although the Conner-Patterson model [Conner 1982] is meant to describe how a proposed change is proceeding (or failing to proceed) at the organizational level, it is important to note that the individual is still the unit of change. Organizations cannot and do not change; only individuals can and do change, and the change cannot become pervasive across the organization unless a critical

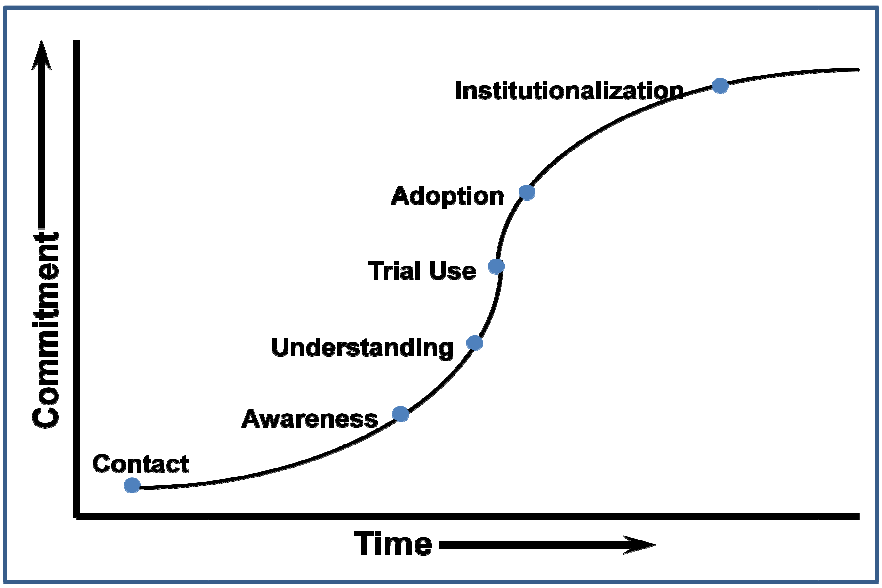

Figure 6: The Conner-Patterson Model of Technology Adoption in Organizations

mass of the affected individuals willingly accept and adopt the change. The phases of the model describing organizational commitment to a technology change are as follows.

- **Contact** – Individuals in the organization gain an initial awareness that a change is being considered or planned for implementation.
- **Awareness** – Individuals know that a change is planned, but lack information about details such as the nature of the change, the extent of the change, and how they will be personally affected or what roles they will play.
- **Understanding** – Individuals comprehend the nature of the change, including specific alterations to work processes, and know the roles that they will be expected to fill during and after the change.
- **Trial Use** – Selected groups of individuals receive necessary training and begin to implement the changes in their normal work.
- **Adoption** – Most or all of the individuals in the organization openly demonstrate their willingness to accept and embrace the change.
- **Institutionalization** – individuals in the organization accept the change as the new status quo, and would actively resist any attempts to revert to earlier methods or processes.

3.1.4 Categories of technology change adopters

In any introduction of a new technology, process, or methodology, some individuals are naturally more willing to adopt the innovation than others. Rogers [2003] noted that in any social system into which an innovation is being introduced, individuals fit into one of five categories according to their willingness to consider and adopt the innovation. Rogers also notes that individuals may change adopter categories depending on the nature of the innovation being introduced and the environment into which the change is being made. These categories are depicted in Figure 7 [Rogers 2003].

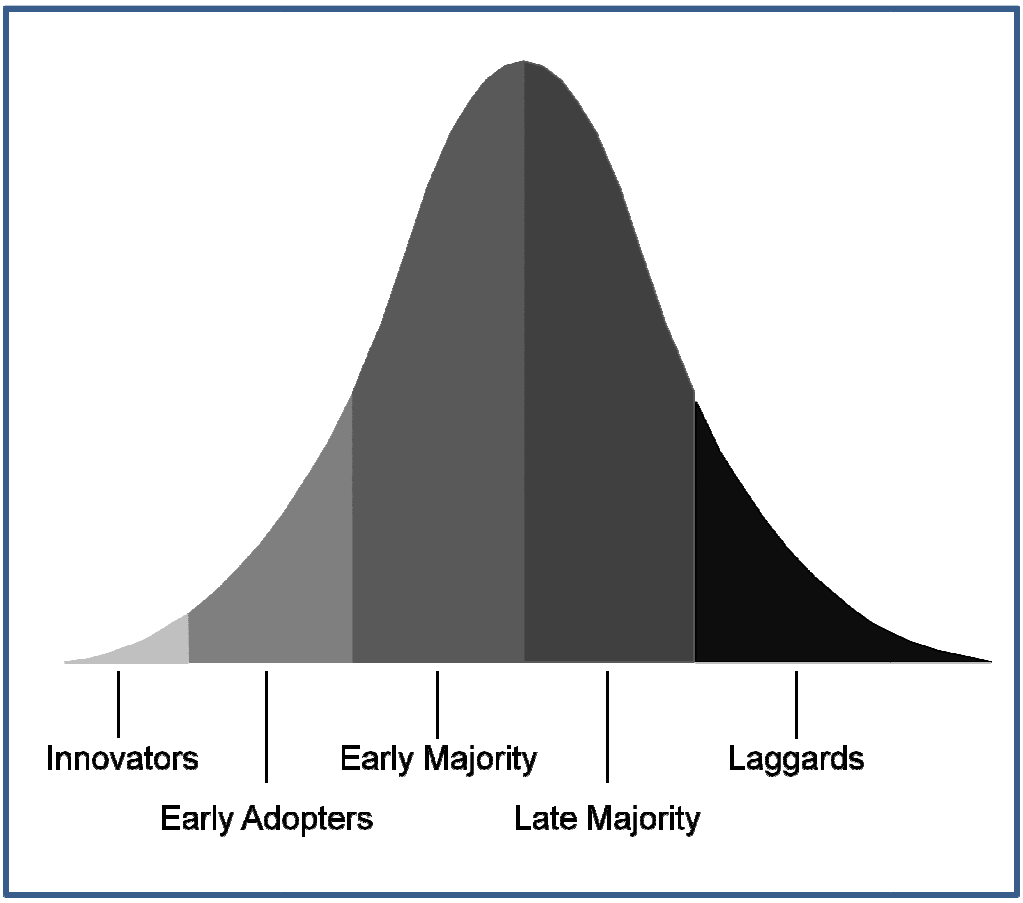

Figure 7: Adopter Categories as Defined by Everett Rogers

When organizations are introducing an innovation or technology change, the individuals in charge should take the adopter types into consideration and focus their initial introduction efforts on respected members of the innovator and early adopter categories.

- **Innovators** are the first individuals to adopt a new technology. These individuals are educated, have multiple sources of information about the innovation, and are most likely to take risks. They appreciate technology for its own sake and are motivated by the idea of being a change agent in their social group. They are willing to tolerate initial problems that may accompany new products or services, and are willing to attempt to find solutions to any problems encountered.
- **Early adopters** are the visionaries in their market who are looking to adopt and use the new technology to achieve a significant competitive advantage in their industry. They have the highest degree of opinion leadership in their particular environment. Early adopters are typically younger in age, higher in social status, and more socially forward than late adopters. They are attracted by high-risk, high-reward projects and are not very price sensitive because they envision great gains in competitive advantage from adopting a new technology.
- **Early majority** individuals adopt new ideas slightly faster than the "average" member of their social system. These people interact frequently with their peers, but typically do not hold positions of opinion leadership. Early majority adopters may deliberate for quite some

time before making a decision to adopt a new change, but once convinced, they willingly follow the adoption trend. Because of their position between the first adopters and those who resist change, they provide a vital connecting link in their social system's interpersonal communication network. Without the buy-in of the early majority, an innovation adoption is unlikely to achieve the necessary critical mass required for a self-sustaining introduction.

- **Late majority** adopters are skeptical, traditional, and less willing than the "average" member of their social system to adopt new innovations. Often, they adopt only reluctantly, due to overwhelming peer pressure. Almost all of the problems or uncertainties of an innovation will have had to be addressed before the late majority will consider adoption.
- **Laggards** are technology skeptics who want only to maintain the status quo. They tend to lack any status in their social system and typically are isolated from their peers. They must be convinced that an innovation cannot possibly fail before they will consider adopting it.

Knowledge Area 3.2: Piloting TSP in an Organization

This knowledge area describes the general requirements and guidelines for initiating and conducting TSP pilot projects.

3.2.1 The TSP pilot

Successful introduction of TSP – or any new technology – into an organization is usually best accomplished by conducting one or two small-scale pilot projects, rather than introducing TSP into an organization across numerous projects and divisions. Pilot projects are required for successful TSP introduction because adoption of the TSP requires strong management support and understanding, and it takes time to acquire this support and understanding. By concentrating the introduction effort on one or two projects, rather than spreading the effort over a wider area, the chances of the pilot projects' success is increased. Broad TSP introduction is generally not possible until there is enough compelling evidence to convince the organization's general population that the TSP methods are effective for the target organization. A series of successful pilot projects can provide this evidence. Furthermore, because organizations have unique needs that may require some tailoring of the TSP applications, a gradual introduction allows the external coaches sufficient time and experience to work with the organization sponsors to learn how the TSP introduction should be adapted to best meet the organization's needs. Early trials also allow coaches and managers to identify potential TSP coaches from within the organization and ensure that they are trained and mentored before scaling up the organization's introduction efforts.

TSP pilot introduction usually involves the following steps.

1. Obtain a management sponsor for TSP implementation.
2. Identify external and internal resources for teaching PSP and TSP skills and for coaching the pilot teams.
3. Provide PSP/TSP instructor and coach training to internal resources.
4. Train top management.
5. Select two or three initial projects or teams.
6. Train the selected teams and their managers.
7. Launch the teams.
8. Monitor the projects and make adjustments as needed.

9. Expand the scope, selecting additional projects or teams.
10. Create or expand the pool of available instructors and coaches.
11. Repeat the process, starting at step 5.

3.2.2 Selecting TSP pilot organizations

There are five fundamental characteristics that must be present in an organization's culture to empower successful adoption of the TSP. Without these elements, the organization is unlikely to implement the self-directed teamworking practices essential for successful TSP introduction. These characteristics should be considered as essential guidelines for selecting an organization in which to pilot the TSP.

- **Honesty** – Both management and workers should value genuine, fair, and unbiased assessments of the organization; people must be willing to face facts and must be comfortable with admitting weaknesses and errors.
- **Rational management** – Managers at all levels of the organization must be willing to listen to working level issues and concerns, and base decisions on information, facts, and data.
- **Disciplined commitments** – The organization's managers and personnel must recognize that all unplanned commitments are unrealistic and likely to be missed; they must accept the need for and value in making rational commitments based on realistic plans.
- **Commitment to quality work** – The organization's managers and employees must believe that using the best available methods for doing a job is always the fastest and cheapest way, and that using best practices will always produce the best products.
- **Trust** – Management must be willing to entrust the teams with the information needed to perform their jobs and must allow the teams to plan and manage their own work.

3.2.3 Selecting TSP pilot projects

In selecting pilot projects, it is generally wise to choose important projects that will be representative of the organization's work, that will provide good reference points, and that are unlikely to be cancelled. The following guidelines should also be considered when choosing potential pilot projects.

- Select project teams that have a supportive management chain. Avoid projects where any manager in that chain who has not been TSP trained or does not fully support the introduction and use of the TSP.
- Ensure that all of the TSP team members are willing to be trained and to try the TSP on their project. It is normal for one or more team members to be skeptical, but it is best to avoid piloting TSP in a group where even one team member is strongly opposed to trying the TSP. When possible, include members who can be effective ambassadors to the most important groups for subsequent TSP introduction.
- Select projects that are reasonably early in the development process. Ideally, the project should be in the requirements or early design stage. Initiating a TSP project at the end of the design phase is acceptable. Projects that are completing implementation or are already in the test phase should be avoided, since work patterns – and defects – have already been engrained and the effectiveness of TSP methods will not be evident in these projects.
- If possible, pick projects of six to nine months duration so that results can be seen and publicized relatively quickly.

- Select projects that are of a reasonable size. Projects with less than about five members are typically too small to provide useful data, and projects with more than a dozen or so members are usually too large for inexperienced team leaders and management to handle properly, at least in the initial stages of TSP introduction.
- It is generally wise to avoid projects that are so troubled or under such heavy schedule pressure that first-time TSP teams would have trouble maintaining process discipline.
- When making the final selection, choose at least two – and preferably three or four – projects in which to pilot TSP. Even in the best organizations, projects get cancelled or redirected. If only one or two pilot projects are chosen and one or both are cancelled, the TSP effort will be delayed or even completely destroyed. In addition, each pilot project provides evidence of the effectiveness of TSP in the organization. If all of the pilot projects are successful, the organization has more evidence to support the initial management decision for introducing TSP, increasing the probability of obtaining the additional support required to expand the TSP. Each pilot should also be viewed as a training program for team leaders and potential coaches since these skills will be in short supply when TSP is expanded to other projects; several early pilots will increase the pool of candidate team leaders and internal TSP coaches.

3.2.4 Selecting TSP champions

Human beings are notoriously stubborn when it comes to accepting changes to their normal habits, even when the changes are obviously beneficial. Therefore, when implementing organizational changes - such as process improvement methods like TSP – the change is more readily accepted if it is promoted by respected members of the organization. These individuals act as *champions*, or advocates for adopting and using the desired change [Rogers 2003]. The TSP coach should work with senior management to choose an appropriate champion whose job will be to maintain management's focus on TSP introduction and who is empowered to implement the change under the guidance of a TSP coach. The champion becomes the individual in the organization who is responsible for implementing and managing the TSP pilots.

The champion and coach must work together to ensure the success of the TSP pilots. The coach should assist the champion in establishing an implementation plan and reviewing that plan with the affected product managers. The coach supports the champion in implementing the plan and coordinates with the champion regularly to monitor progress against the plan.

3.2.5 Implementing the TSP pilot

When implementing organizational change – such as a TSP pilot – the change is more readily accepted if it is actively and consciously managed by the organization, starting at the highest level of executive management. By managing the change, the organization increases the probability of success and minimizes any short-term productivity dips that may result from the implementation of new work processes. To effectively manage organizational change, the responsible parties must

- create awareness of why the change is happening
- build a desire in members of the organization to support and participate in the change
- provide the knowledge needed to make the change
- demonstrate ability to implement new skills and behaviors
- provide a reinforcing environment that will sustain the change

The change management activities outlined above can be easily implemented by the organization's executives through the simple action of defining precisely what objectives they want to achieve through implementing the TSP effort and then sharing those goals with the other members of the organization. The executives must also maintain a visible commitment to and focus on those goals to convince all of the involved people that the TSP effort is an important goal. That commitment is best demonstrated by asking for and publishing regular progress reports of the TSP effort, and by publicly acknowledging and rewarding pilot teams' successes. Other individuals in the organization also have responsibilities for implementing TSP during the pilot, as follows.

- The line managers should be held responsible for meeting the specific goals of building and maintaining their TSP teams.
- All TSP teams should have the goal of faithfully following the TSP process.
- Each individual on TSP teams should have a personal goal of using the PSP (for software developers) or equivalent best practices (for non-software developer team members).

Knowledge Area 3.3: Preparing Management and Teams for TSP Implementation

This knowledge area describes the training and pre-launch preparations required to prepare executives, managers, team leaders, and team members for effective participation in TSP implementations.

3.3.1 Preparing executives for TSP implementation

The *TSP Executive Seminar* generally provides sufficient training for senior or middle managers to oversee implementation efforts or to attend and effectively participate in the appropriate portions of a TSP team launch.

3.3.2 Preparing managers for TSP implementation

The immediate managers of TSP team leaders and any executives or managers who will make presentations in launch meeting 1 or who will approve the team's plans in meeting 9 should do the following.

- Attend *TSP Executive Seminar.*
- Read the launch preparation materials, including the product and senior management goals discussion guidelines.
- Receive instruction on the purpose and content of TSP launch meetings 1 and 9 and management's role in those meetings.
- Seek guidance from the coach or the team leader in preparing the meeting 1 presentation.
- Ask for feedback from the coach and the team leader on a draft of the meeting 1 presentation to ensure that the message is clear and addresses the information needed by the team to make a plan.
- Seek guidance from the team leader or coach on establishing cost, schedule, and quality goals for the team.
- Receive instruction in principles and methods of building, leading, and supporting self-directed teams.
- Be trained in assessing and reviewing team plans and (re)launch products.

3.3.3 Preparing team leaders for TSP implementation

TSP team leaders must attend TSP team leader training before participating in the launch with the TSP team. When possible, team leaders should participate in management preparation meetings, discussions and training. Team leaders also must have a working knowledge of the following.

- All of the materials in the team leader portion of the launch preparation guidelines
- How to manage to the team plan
- How and why to maintain a consistent product focus
- Reasons and methods for following the team's designated processes
- How to address and resolve common team-member process problems
- How to manage product quality and process quality
- How and when to get assistance or support from higher management
- When and how to report team status to management
- Why and how to protect the team
- Strategies to use in fostering team members' professional development
- Reasons and methods for motivating continuous team improvement

3.3.4 Preparing team members for TSP implementation

All TSP team members must have received PSP training (for software developers) or a TSP/PSP introductory course that includes training in defining and following a personal process (for non-software professionals on the team). TSP team members must know these specific techniques.

- Gathering process data on their personal work
- Using personal data to plan and track their work
- Using personal data to improve their work processes
- Making quality plans and gathering quality data
- Using personal quality data to manage the quality of their work

The TSP team leader or coach should ensure that the team members complete the following tasks before the team launch.

- Read the team member portions of the launch or relaunch preparation materials.
- Obtain and read any available requirements documentation pertaining to the team project.
- Familiarize themselves with any technology relevant to the project, and, if possible, become familiar with the experiences that other projects in the organization have had with these technologies.
- Obtain any relevant historical data available as pertaining to size, productivity, and quality on similar efforts, at both the individual and team levels.
- Identify any potential controversial issues and formulate a plan for handling them during the team launch; the team should collectively decide if and how to ask about such issues during launch meetings 1 and 9.
- Perform any requested preliminary work requested by the team leader or coach to produce a high-level conceptual design for the project.

3.3.5 Preparing team members for using TSP support tools

Tool support is essential for the effective and efficient use of the TSP process. Except for very small projects, there will be far too much data to record manually. In addition, the tasks of sorting, tracking, and reporting on recorded data is simply too complex and voluminous to handle without

automated support. Teams should not attempt to use an unsupported TSP process. All teams occasionally need professional tool support, but first time TSP teams must have assistance from a coach or other experienced professional who understands the team's selected support tool and is able to use and explain all of the tool's functions.

Any tool chosen to support the data collection and analysis activities of the TSP team should support the following essential functions.

- Setting up and distributing copies of the support tool for each team member
- Defining processes
- Creating quality plans
- Making statistically sound estimates
- Defining defect and data types
- Recording time, size, defect, and task completion data
- Summarizing time, size, defect, and task completion data and generating reports
- Performing data analyses
- Creating individual team member plans and consolidated team plans
- Revising plans
- Tracking individual and team performance
- Recording and tracking risks and issues
- Generating individual and team project status summary reports

3.3.6 Scheduling the TSP launch

The initial team launch should be scheduled to occur as soon as possible after all of the involved managers, leaders, and team members have been trained, as dependent on the availability of the coach and suitable launch facilities. Ideally, the launch should occur in the early phases (requirements or design) of the project. Other factors to consider when establishing a launch date include the following issues.

- If the team is launching a new project, senior management (or a manager higher than the team leader) <u>must</u> be present in meeting 1 to explain the need for the project and define the management goals. The same manager must also be present in meeting 9 to accept or reject the team's plan, or to negotiate acceptance of an alternative plan. Whoever attends these meetings as the management representative <u>must</u> have authority to approve the project and the team plan.
- The team leader <u>must</u> be present and able to participate in the entire launch.
- The team members <u>must</u> all be present and able to participate in the entire launch, particularly if launching for the first time. Allowances may be made in cases where a team member has an extreme emergency (health or similar issues) and must be absent for some or all of the meetings.
- Development projects should not be launched if the business justification for the project has yet to be finalized, or if the project is over halfway through the implementation phase.

3.3.7 Scheduling TSP relaunches

The next team relaunch should be planned during each team launch or relaunch. Factors to consider when establishing a relaunch date include the following issues.

- The relaunch should be scheduled well before the planning horizon for the current plan will be reached.
- Even if the planning horizon has not been reached, a relaunch should be scheduled whenever the team concludes that its current plan is no longer providing useful guidance for the work. If only one or a few team members need to adjust the schedule, or if the team workload needs to be rebalanced, a replan (rather than a full relaunch) may be adequate. Relaunches should be scheduled only if required changes are too important or too extensive to be addressed in a replanning session.
- A relaunch should always be planned at the earliest possible time following a change in the project goals, major changes in project requirements, a substantial change in team membership, or if the team leader leaves the team and is replaced by someone outside the team.
- As with a team launch, relaunches should be held only if the team leader and most or all of the team members can be present for the entire relaunch. Exceptions can be made in case of urgent need for the relaunch or if an individual team member is unexpectedly unable to attend.

Knowledge Area 3.4: The TSP Launch Meetings

This knowledge area provides an overview of the TSP launch and a description of each of the meetings that make up a TSP launch.

3.4.1 Overview of the TSP launch

The TSP launch is a structured series of team activities guided by a TSP coach; the launch process lasts from two to five day, and includes nine meetings and a postmortem. During the launch, the team learns from management what it is supposed to do, makes a plan for doing the desired work, and then reviews the plan with management. The two desired outcomes of the launch are an approved team plan for producing a particular product, and a jelled self-directed team.

- Each launch meeting has an agenda and a script that the TSP coach and team leader use as a guide to help the team complete the agenda for that meeting.
- Each meeting starts with two team members agreeing to take the roles of recorder and timekeeper.
- Each meeting has a series of designated activities, and each activity is led by the individual who is responsible for maintaining the team focus on the specific topics associated with that activity.
- Each meeting ends with the recorder reviewing the significant decisions and actions resulting from the meeting.

The TSP launch meetings and the major meeting activities are as follows.

- **Meeting 1** – Management presents the project objective and business goals, and describes any critical cost, schedule, or quality requirements. The team also learns about marketing factors such as key competitive issues that might affect the product's design. The team has a chance to ask questions about the requirements and constraints in order to clarify their understanding of the management presentation.
- **Meeting 2** – The team defines its goals and selects team manager roles.

- **Meeting 3** – The team creates the conceptual design for the product, defines the team's process, defines the process and support plans, and produces the project strategy and work breakdown structure.
- **Meeting 4** – The team estimates the sizes of the conceptual design's principal parts and produces the overall plan with total project resource needs and the development schedule.
- **Meeting 5** – The team produces the quality plan.
- **Meeting 6** – The team produces detailed personal plans for the next project phase, or cycle, for each team member, combines these plans into the overall team plan, and balances the team workload.
- **Meeting 7** – The team identifies and mitigates risks associated with the plan.
- **Meeting 8** – The team prepares a briefing of its plan to present to management.
- **Meeting 9** – The team presents its plan (or alternative plans) to management, takes questions about the plan, and receives approval from management for the plan or an alternative. In rare cases, if management does not like any of the alternative plans, the team may be asked to restart the launch and produce additional alternatives.
- **Launch postmortem meeting** – The team discusses the launch process, the performance of all parties in the launch (coach, leader, team members), and other pertinent topics, and generates process improvement proposals and a launch evaluation.

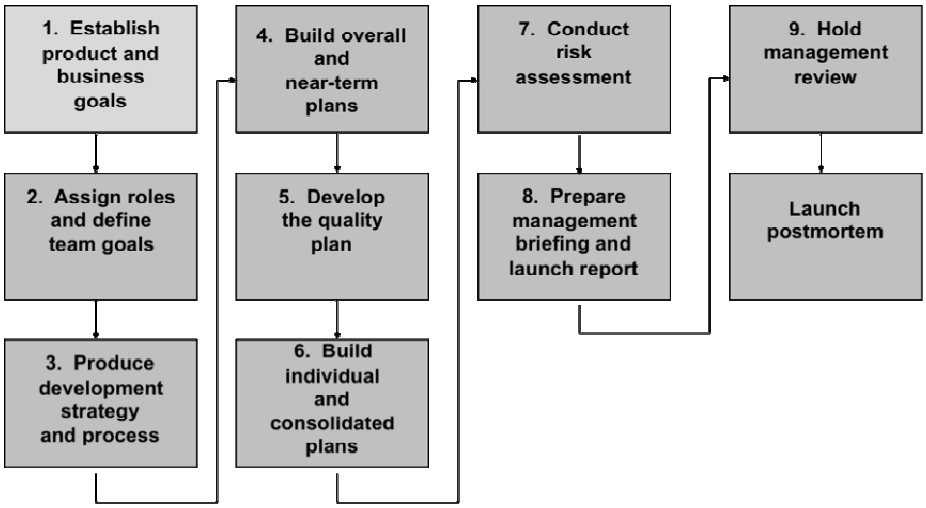

Figure 8: The TSP Launch Meeting Sequence

3.4.2 Launch meeting 1

The opening management meeting is the first step in the TSP launch process. The senior manager describes the business goals, the need for the project, and any critical cost, schedule, or quality concerns. The marketing manager describes the product to be developed, how it is to be used, and the key competitive issues that might affect the product's design. By the end of the meeting, the team should understand the project requirements, the available resources and constraints, and should make sure that any questions or concerns are addressed. In particular, they should make sure that they understand

- the characteristics of the desired product from both a business and customer perspectives

- what is essential, what is optional, and why
- management's criteria for determining success
- additional resources that may be available to the team
- the degree of the team's flexibility with the project schedule, team staffing, and product features

3.4.3 Launch meeting 2

In launch meeting 2, the team defines its goals and agrees on the members' role manager assignments.

The team's goals define the objectives and motivate the work of all of the members. The goals provide the team members with a mutual understanding of what they are expected to achieve and establish a common framework for project planning. Goals

- should be quantitatively measureable whenever possible
- establish a baseline against which to measure progress
- provide motivation in the form of timely feedback about performance and achievements

During the goal-setting portion of meeting 2, the team members begin by listing any explicit goals stated by management and determining what, if any, implicit goals were communicated during the presentation. The team uses these goals to formulate their own goals for the project, ensuring that whatever goals they set for themselves are in alignment with management's stated and implied goals. Team members are each assigned responsibility for tracking one or more goals as the project progresses, and are expected to make periodic reports on status towards meeting the goals.

During the role selection portion of meeting 2, the team members choose individuals to fill each of the designated manager roles (see Knowledge Area 2.5) and assign a back-up manager for each role. If the team determines the need to create positions to supplement the eight standard roles, the team members define and fill those role positions as well; however, new role manager positions should be defined sparingly and with caution (see Knowledge Area 2.5.12).

3.4.4 Launch meeting 3

TSP planning starts in launch meeting 3, which provides the context for the plan through the following activities.

- Producing a conceptual design that defines the principle components, elements, or features
- Establishing the project strategy
- Defining the work products to be produced for the next phase or cycle, and all subsequent phases or cycles
- Agreeing on the development process
- Creating the process and support plans
- Defining the tasks and weekly reporting for each management role

The *conceptual design* defines the preliminary design approach for the planned product and names its principal elements and their functions. The sole purpose of the conceptual design is to identify the product's principal elements and functions so they can be used in making the size estimate. Product plans are based on conceptual designs because the effort required to develop a work product is closely correlated with that product's eventual size. An accurate size estimate

enables the team to make an accurate estimate of development time. Because PSP data show that products developed from the same requirements can vary in size by ten times or more depending on the design approached used, the accuracy of the size and time estimates must be based on the design approach that will be used in developing that product. And, since size and time estimates typically are needed before development starts, it is essential to produce a conceptual design for use in product planning. The general strategy for producing a conceptual design consists of five steps.

1. Define the product's principal functions.
2. Postulate a small number of parts (about 10 or 15) that when used together will perform the product's intended functions.
3. Estimate the sizes of these parts by comparing them with existing parts of known size.
4. If some of the parts are too large or complex to estimate directly, consider subdividing and estimating them as in step 3 above.
5. Use the sum of the part sizes to generate the size estimate for the product.

The team's *product strategy* is produced after completing the conceptual design. The team must decide how to subdivide a product(s) into parts that can be separately developed, how to put these parts together to produce the finished product(s), and how to test the completed product(s).

After producing the conceptual design and product strategy, the team defines the *work products* to be produced for the next phase or cycle, and in all subsequent phases or cycles. This is generally done by creating a work breakdown structure (WBS) that decomposes the project into its component elements and provides clarification of the project deliverables or tasks. Each of these components can and should be separately planned, since the overall plan is more accurate if it is based on the total of the plans for the separate components rather than on a single large plan.

When making the WBS, the team must decide whether to base the WBS on products or tasks. A *product WBS* concentrates on deliverables, and produces a list of distinct and separate work product. A *task WBS* defines all of the tasks needed to create the product and its components, starting at the macro level (phase, major task area, or task groups) and the breaking down the macro level tasks to produce a detailed list of the discrete work elements performed by each individual team member.

Once the team has defined what it needs to build, it must agree on the *development process*, which defines how the team will do the work, with a focus on doing the job in the most effective and efficient way. The team members consider the products they will produce and the issues they are likely to encounter. The team should also consider the lessons learned from prior projects and incorporate these lessons into the process and plan for this job. The product of this step is a script that is sufficiently detailed to guide the team members' task planning and project work. Once the team has produced the process script, the process manager should document it and place a copy in the project notebook.

The final steps of launch meeting 3 are to create process, support, and role plans. The *process plan* is for developing any additional process elements that will be needed during the currently planned project cycle. The team also identifies any missing or needed development or support tools and facilities; the support manager documents this list and the team makes a *support plan* for

developing or obtaining the needed tools and facilities. The *role report* defines needed tasks for each management role and specifies what should be reported by each role manager at the weekly team meetings.

3.4.5 Launch meeting 4

In meeting 4, the team produces the overall plan, determines the overall project resources needed, and establishes the development schedule to project completion. The meeting 4 activities include the following.

- Estimating the size of each item identified in meeting 3.
- Producing a detailed task plan for the near term and a high-level plan for the remainder of the project.
- Estimating the resources required for each defined task. The total project resource estimate is the sum of the efforts estimated for each sub-task, or work product. The total must include sub-task or work product estimates for overall product design and architecture, project coordination, project management, component integration, and system testing.
- Estimating the aggregate team task hours available for each project week. All of the team members estimate the number of task hours they expect to have available for each week of the project, subtracting time for vacations and non-project commitments.
- Generating the size and time estimates, and producing the team plan for the entire project. The team plan is generated by calculating the calendar time required to complete all of the required tasks, and identifying and synchronizing inter-team and intra-team dependencies.
- Reviewing the plan to ensure that it follows the defined strategy, produces the required products, meets the established team goals, addresses all critical internal and external dependencies, and includes tasks for every defined product.
- Developing alternative plans if the first plan does not meet all cost, schedule, and functionality goals. The alternatives may be based on adding more resources (taking into account the need for training those resources), producing the product in several iterations, or reducing functional content.

3.4.6 Launch meeting 5

After completing the overall plan, the team produces the quality plan for the work to be done. The members agree on measurable quality goals, the actions they will take to achieve those goals, the quality commitments that they will make to management, and the team members' responsibilities for managing and tracking these quality commitments. Meeting 5 activities include the following.

- Reviewing the team's quality goals from meeting 2
- Listing the team members' quality activities
- Estimating the number of defects to be injected in each project phase
- Estimating the defects that will be removed in each project phase
- Calculating the total number of defects to be left in the product at the end of each process phase, for the total product and for each product assembly
- Calculating the various required quality values, ratios, and densities
- Evaluating and adjusting the plan until the team's quality goals and schedule are met. The team may need to evaluate the resulting defect densities and removal rate, make needed adjustments in phase times and yields, or recalculate the quality parameters.
- Documenting the quality plan

3.4.7 Launch meeting 6

In launch meeting 6, each team member produces a detailed personal plan for the next project phase, or cycle. The team members perform the following activities.

- First, as a group, decide which tasks to perform during the immediate next project phase, or cycle, (a few weeks or months) and then allocate each next-phase task to one or more team members.
- Make individual plans for accomplishing the tasks allocated to them in the next phase, or cycle. As needed, they break each task down into sub-tasks of ten or fewer task hours. Larger tasks that occur in later phases, or cycles, do not need to be estimated at the sub-task level.
- Combine the detailed individual plans to generate the consolidated overall team plan.
- Balance the workload as needed so that they all finish their task work at approximately the same time, in order to minimize project costs and keep the schedule as short as possible.

3.4.8 Launch meeting 7

Risk management is important because it is more expensive to recover from a risk that becomes reality than it would have cost to avoid the risk in the first place. Many risks can be predicted in advance and avoided. Therefore, risk identification and mitigation is included when making the team plan during the launch. During meeting 7, the team conducts a project risk assessment by performing the following activities.

- Identifying potential risks to the project work. The TSP defines a risk as something that may or may not happen; events that are certain to occur are *issues* and should be addressed in the team's plan.
- Evaluating the likely impact of each risk as "high," "medium," or "low" according to the consequence to the schedule: a high-impact risk would normally delay the project by a few months to a year or more, a medium risk would have a delay of a few weeks, and low-impact risk would delay the project by a few days. This rank range may need to be adjusted for very long or short duration projects.
- Judging the likelihood of occurrence of each risk as "high," "medium," or "low"
- Prioritizing risks in order as high-high high-medium, medium-high, or medium-medium. Risks that are rated as low-medium, medium-low or low-low are not ranked. Each ranked risk is assigned to a team member for tracking. The team should decide how often each risk should be reviewed and establish a schedule to review it. The team member assigned to track that risk should then set a follow-up schedule for reviewing the risk with the team.
- Determining an effective mitigation plan for all near-term risks. Risk-mitigation strategies are typically of three types.
    1. If the risk is not under the team's control, such as specification approval or available system test facilities, the mitigation action is to get the assistance of the responsible group or to appeal to management for help.
    2. When the risk concerns the team's internal work, the mitigation action would typically require making an alternate plan to follow were the risk to materialize.
    3. When the risk involves the joint action of several groups, a mitigation team should be formed with members from all the involved groups. This team devises a mitigation plan using either strategy 1 or 2.

- Planning a schedule for reviewing each risk being tracked. During the follow-up review, the team will decide if mitigation is required, when the next follow-up review should be held, or if the risk should be dropped and no longer tracked.

3.4.9 Launch meeting 8

In meeting 8, the team prepares to present its plan to management in meeting 9. In the plan presentation, the team should cover the meeting agenda and objectives, a management overview, summary of work done, team goals, team roles, the plan, alternate plans, risks, and questions and discussion.

3.4.10 Launch meeting 9

In meeting 9, the team leader presents the team's plans to management with the entire team there to offer support and answer questions. The team may be called upon to defend its plan to management. The points that should be made when defending a plan are as follows.

- This is the best plan that the team can make to meet the stated needs.
- If the requirements or resources can be changed, the team will reexamine the plan to see how the changes would affect the schedule.
- If the plan needs to complete the work at a lower cost or on a shorter schedule, the team could develop additional alternate plans.

At the conclusion of the meeting, management generally approves either the team's suggested plan or an alternate team plan. If management is not satisfied with any of the alternatives, they may ask the team to consider other alternatives and to come back with a revised plan.

3.4.11 Launch postmortem

The postmortem provides a structured learning vehicle for the team, the team leader, and the coach. It is where the team provides feedback on the launch process, the team leader's performance in helping the team during the launch process, the coach's effectiveness in guiding the team and supporting the process, and any other topics that the members feel are important. The team documents all process improvement proposals (PIPs) and also completes the launch evaluation forms.

Knowledge Area 3.5: The TSP Relaunch

The team relaunch is nearly the same as the launch except that it is done by a team that has already completed an initial launch of the same project. This knowledge area describes how the relaunch differs from the launch, explains when and how to conduct a relaunch, and delineates the inputs and outputs for the relaunch.

3.5.1 The relaunch

A relaunch is much like a launch in that its primary purpose is to produce a detailed plan for the next cycle. In an ideal world, the project would have proceeded exactly as originally planned so that the only work that the team would need to do in the relaunch is to produce and balance the plans for the next phase or cycle. However, the reality is that projects rarely proceed entirely as planned, so teams generally have to adjust and refine their overall plan.

The most significant difference between a launch and a relaunch is that the team members have already worked together. In a relaunch, the team members update their plans based on what they have done since the initial launch or the prior relaunch. Another important difference is that the team has already committed to management what it intends to do and, if that commitment is unchanged, the members do not need to repeat the management meetings (launch meetings 1 and 9). However, if the project has changed in any significant way (such as changes to the product requirements, the team membership, project schedule, project scope, etc.), then the relaunch should be regarded as a new project launch and all of the meetings and activities should be held as described in the sections of the BOK covering launch meetings 1 through 9. Otherwise, the relaunch process should proceed as described below in section 3.5.3.

3.5.2 When to hold a relaunch

There are several occurrences that usually signal the need for a relaunch. These include reaching the end of a planning horizon, completing a project phase or cycle, adding or losing team members, or incurring major changes to the project.

- **Reaching the end of a planning horizon** – during the initial launch, teams make detailed plans only for the following next few weeks or months. Because of the normal fluctuations of the work, periodic plan readjustments are necessary to rebalance the workload and to address any other changes. Teams often include relaunches as part of their initial plan, either when a certain calendar date is reached or when a project phase or cycle end date is scheduled to occur.
- **Completing a project phase or cycle** – Many teams plan relaunches at the end of a project cycle or phase. However, it is important to take into consideration that the needs of the team may not correspond precisely with the scheduled phase or cycle end, and understand that some work may end early or bleed into the next cycle. Therefore, if a relaunch is needed before the end of the phase or cycle, it should be scheduled as needed, rather than requiring the team to wait and continue to use plans that are outdated or no longer useful.
- **Adding or losing team members** – When teams lose members, the work allocated to those individuals must be redistributed among the remaining team members.
- **Incurring major changes to the project** – Projects frequently change: requirements change, markets change, or the team learns more about the desired product. When projects change enough to affect the team's commitments to the customer or to management, the team should hold a relaunch. If the changes are sufficient to render the team's current plan obsolete, the team should hold a completely new launch.

3.5.3 The relaunch meetings

A typical relaunch is much like a launch, except that the team does not hold the overview meeting with management and customer stakeholders (launch meeting 1), the outbrief planning meeting (launch meeting 8), or the outbrief to management and customer representatives (launch meeting 9). There are seven relaunch meetings and a relaunch postmortem. Except for meetings 1 and 5, the relaunch meetings are the same as the corresponding meetings in the launch process.

The differences in the relaunch meetings are described below.

- **Relaunch meeting 1** – The first step in the relaunch meeting is an overview of the project status as compared to the plan. The team leader reviews the work that has been completed and reviews the team's progress against the planned goals. If there are any changes to the management goals for the project, the team leader explains why the changes were made and how success against the revised objectives will be measured. Management or customer representatives do not need to attend this meeting, but interested managers may attend if they would like an updated on the project status.
- **Relaunch meeting 5** – As in the initial launch, the team makes its quality plan in relaunch meeting 5, using the same quality-planning process as in the launch. The team uses their historical injection and yield rates for the phases that they have not yet started. For the completed and partially-completed phases, the team will use the actual data on the defect-removal phases that have already been completed to estimate how many defects were likely to have been injected. The team also estimates the phase yields by examining their actual data from inspections, and combines the estimated yield with the known times for the completed phases to calculate the defect injection rates. Once yield and defect injection rates have been estimated for all of the completed and future phases, the team can complete the quality plan in the normal way.

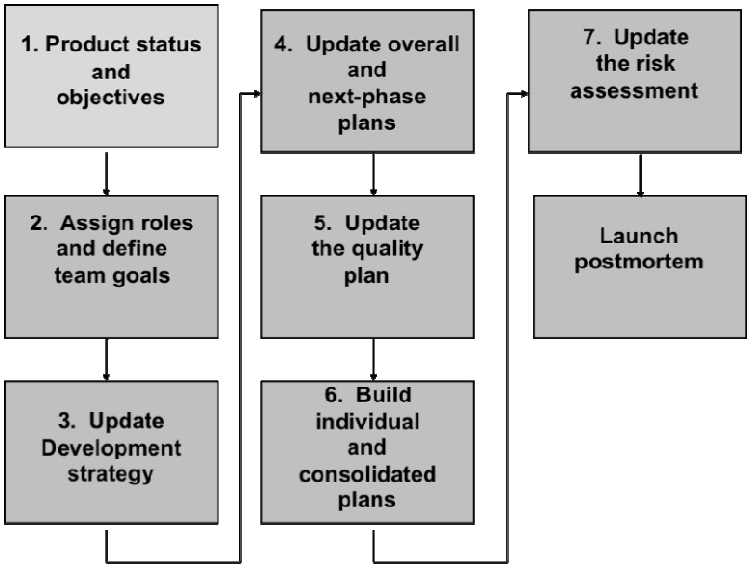

Figure 9: The TSP Relaunch Meeting Sequence

3.5.4 Relaunch inputs

The following inputs are needed for a relaunch.

- Changes (if any) to the project's business needs
- Changes (if any) to management's goals
- Changes (if any) to product requirements
- Trained resources
- Process assets
- Historical data from the prior phase and/or previous projects
- The presence of a qualified TSP coach [Chick 2009]

3.5.5 Relaunch outputs

The following outputs, which are virtually identical to those from the initial team launch, result from a relaunch.

- Revised or reaffirmed quantitative team goals that are linked to business needs, management goals, and product requirements
- Team role manager assignments
- A list of work products for the next phase or cycle
- Cyclic product development strategy
- Size estimate for each work product to be developed in the next phase or cycle
- Tailored process for each work product to be developed in the next phase or cycle
- Task plan for each work product based on the tailored process
- Effort estimate for each task in the next-phase, or cycle, task plan
- Estimated schedule of available resources for the next phase or cycle
- Earned value plan for the next phase or cycle
- Milestones for schedule-based product and business objectives
- Quantitative quality plan for the next phase or cycle
- Quantitative product and process quality criteria for controlling phase entry and exit for each work product
- Risk assessment and mitigation plans
- Assigned resources for every business and product objective, work product, task, goal, and risk identified for the next project phase or cycle
- Overall plan from the beginning of the next phase, or cycle, to the end of the project
- A detailed next-cycle plan
- An empowered and recommitted team

Competency Area 4: Project Implementation and Tracking with TSP

Following a TSP team launch or relaunch, the team members must work to implement their plan by completing the tasks which they have been assigned. The TSP does not mandate the use of any particular approach to project implementation; rather, it stipulates that team members should all use PSP (for engineering work) or best professional practices (for other types of work done in support of the project) to enable team members to produce high-quality product components or project elements. TSP principles also require the team members to observe process discipline by following the processes and strategies that the team agreed to use; by requiring that the team members all record their data accurately, honestly, and in real time; and that the team members individually and collectively maintain a constant focus on quality.

Faithful and precise data collection enables teams to monitor their progress against the plan. Up-to-date and accurate data can be analyzed to enable team members to identify potential problems with project schedule or product quality earlier in the process, rather than later. This allows teams to address these issues, fix the problems, or seek help before small problems become big problems that can derail the project. Using data for status tracking also provides the information necessary to generate reports for its stakeholders on the team's progress, and to advise management of potential problems so that they can provide assistance or additional resources to put the team back on track.

This competency area is composed of the following knowledge areas.

4.1 Weekly Meetings – The weekly team meeting is one of the elements that differentiates TSP from typical software projects. It keeps the team members focused on the project work and allows regular opportunities for teamworking and team building. This knowledge area discusses the TSP weekly meeting.

4.2 Checkpoints – Checkpoints are a series of meetings between the team leader, team members, and coach that allow everyone to discuss issues or problems such as process implementation, data gathering, or data analysis, and to generate solutions for any identified problems. This knowledge area discusses the checkpoint objectives, timing, process, and the content covered in the TSP checkpoint.

4.3 Communicating with Stakeholders – Management is the primary stakeholder of the TSP team, although there may also be internal customers, external customers, or both. In addition, the team members themselves are stakeholders in the process and its outcome. Whoever the stakeholders may be, it is important that the TSP team communicates frequently with them and reports useful and timely information about status, schedule, and product quality. This knowledge area discusses what information to communicate to the various stakeholders and when reports should be made.

4.4 Replanning – Plans are subject to frequent change, particularly on new or inexperienced TSP teams. When plans change frequently it is easy for the team members to lose track of the overall team plan or their personal roles in meeting the plan. To ensure that the team

stays on track and maintains its motivation, it may be desirable to hold periodic short replanning meetings, in which the team leader reviews the current management priorities, and the team members summarize their status and adjust their individual and team plans as needed to address those priorities. This knowledge area discusses the reasons that may necessitate replanning, and describes some of the approaches used by TSP teams to update or revise their plans.

4.5 Phase, Cycle, and Project Postmortems - The postmortem analyzes the collected data on schedule, task hours, estimation, quality, and process. The results are used to provide personal, team and organizational planning data for use in an upcoming launch or relaunch and to identify improvements to process elements. The focus is on using data for planning and process improvement. This knowledge area discusses the postmortem differences, objectives, planning, and processes.

References: The material covered in this competency area is detailed in these primary sources.
[Humphrey 2006a, Chapters 17, 18, 28]
[Humphrey 2006b, Chapters 5, 10, 13]

Knowledge Area 4.1: Weekly Meetings

The weekly team meeting is one of the elements that differentiates TSP from typical software projects. It keeps the team members focused on the project work and allows regular opportunities for teamworking and team building. This knowledge area discusses the TSP weekly meeting.

4.1.1 Weekly meeting activities

The team meets every week to review its status. The team assesses earned value (EV) and task hour status, reviews key milestone status, determines the extent of any schedule exposure and defines a recovery plan if the schedule has slipped, and assesses the status and mitigation plans for the key project risks.

4.1.2 The weekly status report

Each week, the planning manager gathers the basic data required for the weekly status report. These are the task hours planned and actually worked by each team member and by the team as a whole during the past week; the planned and earned value for each team member and the team for the past week; the completed task values; and planned and earned value for the cycle to-date. These data are interpreted and used to prepare reports on the following aspects of team status.

- **Task hours** – Actual hours worked are compared with the planned task hours for team members and the team as a whole to ensure that everyone is getting in the time needed to accomplish the planned work.
- **Planned and earned value** – The planned value (PV) for the week is compared with the actual earned value (EV) for team members and the team as a whole to ensure that the team is making progress against the planned schedule at the established rate.
- **Completed-task values** – The “to-date hours for tasks completed” metric provides the actual hours for the tasks that have been completed to date as compared to the planned

hours for those specific tasks. When the actual hours are above the planned hours, it indicates an underestimate and when actual hours are below planned hours, an overestimate.

- **Planned and actual EV** – If the actual to-date values are lower than the planned to-date values, the team is behind schedule for the entire phase up to that point. If the weekly actual values are lower than the planned values, the team is behind schedule for the week; if both the to-date and weekly values show that the actual are lower than planned, the team is behind schedule and is continuing to slip. When the actual values are above the planned values, the reverse is true.
- **Weekly and to-date EV** – By comparing weekly and to-date data, the team members and team can tell where they stand against their plans, and whether they are catching up or falling further behind. When the weekly data are closer to plan than the to-date data, team performance is improving. However, if the weekly data are not above the plan, the team is not catching up.
- **Baseline values** – The baseline value on the WEEK form provides the total task hours planned at the time the baseline was established (usually at the launch). By comparing the baseline planned task hours with current total planned task hours, a team member or the team can determine the amount and rate of task growth. A large task growth typically indicates that the team's workload is growing or that the work was initially underestimated. Unless several of the larger planned tasks were overestimated and can thereby compensate for the plan's growth, or unless the team's resources have grown to match the increase in scope, the data indicate that the team is likely to have schedule problems before the end of the phase, or cycle.

4.1.3 The weekly team member reports

The team leader and members should prepare individual reports on their member roles and other items that are reported as part of the weekly meeting script. The team members should focus on key topics that the team leader and other members need to know and should keep their reports as brief as possible.

- **Manager report** – The team leader describes any important new developments and project issues, and discusses key issues and actions expected during the coming week.
- **Role reports** – Each role manager reports any significant activities from the last week, any anticipated activities for the coming week, and other pertinent and significant role-related information.
- **Goal report** – The team members responsible for tracking key team goals report any changes or significant issues on the status of the goal(s) for which they are responsible.
- **Risk report** – The team members responsible for tracking key project risks report and significant risk status changes, or any mitigation actions needed or taken.
- **Team member project status** – Team members report on their individual EV and task hour progress against the plan, significant tasks completed, and planned activities for the next week.
- **Team project status** – The planning manager summarizes team EV and task hour status against the plan, and gives a project-completion estimate.

Knowledge Area 4.2: Checkpoints

Checkpoints are a series of meetings between the team leader, team members, and coach that allow everyone to discuss issues or problems such as process implementation, data gathering, or data analysis, and to generate solutions for any identified problems. This knowledge area discusses the checkpoint objectives, timing, process, and the content examined in the TSP checkpoints.

4.2.1 Checkpoint objectives

The TSP checkpoint is an opportunity for coaches to meet with TSP teams to help them identify issues and problems that could limit a TSP team's performance and to recommend actions to address the problems and issues. The objectives of the checkpoint are as follows.

1. Help the team with common problems and issues
2. Encourage the team to assess its process fidelity
3. Determine if the team members are following their processes
4. Ensure that the team is using its data to assess and report on project status
5. Examine the quality data for evidence that team members are planning to do quality work and succeeding

4.2.2 Checkpoint timing

The TSP process calls for a checkpoint review about one to three months after the first team launch and a second checkpoint after the first relaunch. If needed, follow-up checkpoints should be done again a month or two after each relaunch. If problems are identified in any of the areas covered by the checkpoint, these problems must be addressed promptly or the team's performance will suffer.

4.2.3 The checkpoint process

During the launch or relaunch meetings, the team schedules the checkpoint as a task in the plan. The coach observes one of the team's weekly meetings, analyzes the individual and team data collected since the launch or relaunch, and has one-on-one discussions with each member of the team. The coach might ask questions such as the following.

- How well is the team (and each individual) using their defined process, and how is it working for them?
- Are there any problems or issues with the process that should be addressed using the PIP procedures?
- How well is management supporting the team? Is there anything that management could do to better support the team?
- What things are working well for the team (and each individual)? What accomplishments have been made? How well is the team (or individual) progressing?
- What suggestions does the team (or individual) have for improvement?
- How well are the role manager tasks being handled? Are the problems or issues that need to be addressed?

At the conclusion of the checkpoint process, the coach prepares a checkpoint report to communicate the checkpoint findings (see 4.2.5).

4.2.4 Data examined in the checkpoint

In the checkpoint, data are examined at the team and individual levels. At the team level, the coach looks at the following elements.

- Development task status
- Schedule (in terms of planned versus earned value and percent schedule deviation)
- Effort (in terms of planned versus actual task hours and to-date hours for tasks completed)
- Quality (in terms of phase yield, defect injection rates, and defects removed by phase)
- Size
- Goal tracking
- Issue/risk tracking

When analyzing the team data, the coach should focus on team maturity (adherence to process), and call out both positive comments and areas in which the team needs improvement.

At the individual level, the coach examines the each team member's data and ensures that the individuals are accurately logging time and defect data. Specifically, the coach examines the following individual data.

- Effort (in terms of planned versus actual task hours and to-date hours for tasks completed)
- Quality (in terms of personal review rates and yields)
- Schedule (in terms of planned versus earned value and percent schedule deviation)
- Size tracking

When discussing data with individuals, the coach should give positive feedback, ask follow-up questions, and help each team member to identify areas for improvement.

4.2.5 The checkpoint findings report

The coach prepares a report and presents the checkpoint findings to the team. The findings include a delineation of any team problems or issues that concern the team, the team leader, or management and the coach's recommendations for addressing the problems. These recommendations should take into consideration the issues and suggestions raised by the team during the preliminary group and one-on-one discussions. Any data reported should be included only as team aggregates, and team member commentary should be reported without attributing names. In no case should any information be reported in a way that it can identify individual performance parameters or be construed as an audit-like activity.

The team leader and team members should be given a chance to proposed changes or additions to the checkpoint report before it is finalized. If needed, the team should develop an action plan to address issues or problems identified in the findings, and the action plan should be incorporated into the final report. When the report is complete, the coach and the team leader should review the final findings and recommendations with the team, and then the team leader and the coach make a joint presentation of the report to management.

Knowledge Area 4.3: Communicating with Stakeholders

Management is the primary stakeholder of the TSP team, although there may also be internal customers, external customers, or both. In addition, the team members themselves are stakeholders in the process and its outcome. Whoever the stakeholders may be, it is important that

the TSP team communicates frequently with them and reports useful and timely information about status, schedule, and product quality. This knowledge area discusses what information to communicate to the various stakeholders and when reports should be made.

4.3.1 Project tracking reports

TSP teams most commonly use project tracking reports to communicate information to others. The three principal audiences for project status reviews and reports are the team itself (typically in the weekly status meeting), immediate management (typically through weekly to monthly reports from the team leader), and senior management (as required, usually ranging from monthly to quarterly, or once yearly). The process for generating project status reports is as follows.

- Define the report audience and frequency
- Define the report content for each audience
- Obtain data pertaining to project status
- Produce the status report
- Present the status review or report

4.3.2 Frequency of project status reporting

Teams generate their internal status reports on a weekly basis. All of the team members track their task time, the date on which tasks are completed, and report their status to the planning manager on a weekly basis. They also identify any issues that require team or management attention. The planning manager consolidates team status, prepares the summary team status report, and reviews the summary with the team during the weekly meeting. The team determines which identified issues can be handled by the team and which need to be escalated. At appropriate intervals, the team works with the team leader to produce the intermediate or senior management report. Some organizations have regular status review systems; reports should be submitted at the intervals specified in the system. In organizations without such a system, the team should produce the management reports either weekly or monthly, depending on management's preference.

The principle content of the immediate manager's report should include a summary of the overall project status against the plan (including status against key milestones), the team's actual versus planned EV and task hours, the performance against the quality plan, the current estimated project completion date, a summary of other issues or problems and the actions planned to address them, the areas in which the manager's assistance is needed, and any other topics with which the manager should be familiar. If the project is behind schedule or there is other bad news to report, this information should be briefly summarized at the beginning of the report, along with a statement that more details will follow later in the report. The report should also include a summary of the status of action items from the last report and the status against them.

Reports to senior management should include a high-level view of the information contained in the intermediate reports. As with reports to lower management, the senior management report should open with any bad news, then more details later. The report may also contain information about project milestones, the project strategy, the project's status against commitments, actions being taken to address any problems or issues, and changes in status since the last report. These

reports are given at intervals as desired by the senior management, generally on a quarterly basis, although in some organizations, they may be requested as often as once a month or as infrequently as once a year.

4.3.3 Reporting bad news

If a team is likely to miss a commitment, it should report this information as soon as it becomes apparent. They should be prepared to answer questions from management regarding how late the team is likely to be, why the team is late, the actions being taken to get back on schedule, the plan to address any problems caused by the schedule delay, and what help (if any) the team needs in order to meet its revised commitment.

4.3.4 Providing regular status reports

Managers need regular updates on project status and any problems or issues so that they can better manage costs and balance resources across the organization. Regular reports allow them to know if the project is on schedule or if there are problems that could jeopardize the project, the project quality, or the project budget. Reports enable management to get an idea of how well the team leader is running the project and how well the team is performing against its stated objectives, and provides them with information needed to answer questions from more senior management.

Reports are also important to the team. They provide visible proof to both the team and higher management of what the team members and leader have accomplished, providing motivation for the team and promoting the team's accomplishments. Reports show management that the team leader and team are doing a good job at managing themselves, and that they will deliver a quality product on schedule and at planned costs. Reports demonstrate that the team leader is doing an excellent job of leading the team, and that the team members are following sound and disciplined practices. Status reports also help the team members to better understand where their work fits into the business, and keeps team members focused on the work so that they don't get lost in daily details.

Knowledge Area 4.4: Replanning

Plans are subject to frequent change, particularly on new or inexperienced TSP teams. When plans change frequently it is easy for the team members to lose track of the overall team plan or their personal roles in meeting the plan. To ensure that the team stays on track and maintains its motivation, it may be desirable to hold periodic short replanning meetings, in which the team leader reviews the current management priorities, and the team members summarize their status and adjust their individual and team plans as needed to address those priorities. This knowledge area discusses the reasons that may necessitate replanning, and describes some of the approaches used by TSP teams to update or revise their plans.

4.4.1 Why plans change

There are at least six principle reasons that cause plans to change. These are discussed in more depth in Appendix B of this document. The reasons include the following.

- The team made a poor estimate.
- The team's understanding of the work has changed.

- The resources available to do the work have changed.
- The requirements have changed.
- Management's or the customer's priorities have changed.
- The team has completed or nearly completed its currently planned work.

4.4.2 What teams should do when their plans change

Teams should update their plans whenever the team members feel that their current plans no longer help them to do their work. However, it takes time to replan, and changes to the plan make it harder to track progress accurately; therefore, teams should replan their projects if and when they really need to do so. The decision to update the plan should be based primarily on the team's needs and carried out only after weighing the benefits of replanning against the additional costs in time and project overhead.

When teams have poor estimates (over- or under-estimates), numerous unplanned tasks, or changes in the understanding of the project, they can adjust their plans by means of dynamic planning, workload rebalancing, or replanning meetings. Other problems, such as changes in requirements or resources, reprioritized management goals, or reaching the end of a project phase or cycle, require the team to conduct a full relaunch. If the team's commitments are in sufficient peril to warrant a replan or relaunch, the team should immediately alert management, develop a recovery plan, and promptly explain the problem and recovery plan to management.

4.4.3 Dynamic planning

Dynamic planning is the response to the recognition that plans are, by their very nature, made with incomplete knowledge; dynamic plans include planned activities for updating the plan [Douglass 2009]. When TSP teams do dynamic planning, the members regularly adjust their plans to reflect what they have learned about the job they are doing; this helps to keep the plans accurate and guide the team members in doing their work. Inexperienced TSP teams or teams who are just starting new projects may make plan changes at least every week while they are learning the most about the project.

One of the most common problems that requires teams to dynamically replan is when unplanned tasks must be added to the plan. Unplanned tasks arise even if requirements have not changed or management has not reassigned team members to other projects.

The most common cause of unplanned tasks is that the team develops a better understanding of the work that they have committed to do; they may have forgotten to include necessary tasks, or underestimated the scope of a task or the size of a product component. The simplest way to handle unplanned task is to insert new tasks into the task plan whenever they arise. This gives team members a complete record of both planned and unplanned work, and keeps task hours realistically related to the actual work. Furthermore, this approach provides the task-productivity and weekly task hour data needed to make more accurate plans. By tracking the level and type of unplanned work, team members can make progressively more accurate plans, better anticipate future tasks, and lessen the impact of unplanned work. On the other hand, tracking unplanned work means that team members must regularly add new tasks to their task plans, which may result in an unbalanced team workload, schedule slippages, or other risks to the overall team plan. When

unplanned tasks impact individual or team commitments, the issues – both the reasons for so much unplanned work and the resultant schedule risks – must be addressed.

4.4.4 Workload rebalancing

On any team, some members will work faster than others. This means that some members will complete their task work ahead of schedule, and others will fall behind. This is a normal consequence of variations in estimating accuracy, differences in productivity rates, and variable difficulty in the work. During the weekly status updates, teams should identify the members who are ahead of schedule and can handle additional workload, and those who are late and need tasks to be offloaded. If teams do not rebalance their detailed plans when it is clear that the workload has become unbalanced, some members will eventually fall so far behind schedule that they will delay the project.

4.4.5 The replanning session

When the team's plan is still appropriate to the project requirements – that is, there have not been significant changes in requirements, resource, or management or customer priorities – but several team members have changes in their plans that will affect other team members, the entire team should hold a replanning session. The replanning session includes all of the topics covered by TSP launch meetings 4 and 6. In particular, the team needs to update the overall team plan, the team members' individual plans, the weekly task hour schedules, and the task productivity rates.

- **Update the overall team plan** – When making new plans, the team members revise their team and personal plans and rebalance the workload. If the new plans are not reasonably consistent with the baseline plan, they must meet with management and explain the deviation. They must also get management's agreement to the new plan.
- **Update team member plans** – Team members should update their personal plans based on available personal data on the project to date. The most useful data for task and schedule replanning are weekly task hours, task productivity rates, and the number or frequency of unplanned tasks. If the team member's updated plan is not consistent with the team member's baseline plan, or commitment made to the team, he or she must meet with the team and explain why, so that the team can determine its impact to the team's commitments and make any changes required to the team's plan.
- **Update the weekly task hours** – Team members should update their weekly task hour plans based on the historical to-date project data pertaining to the actual distribution of their weekly task hours. The team members should select a task hour rate that they are reasonably confident that they can accomplish, and then strive to gradually improve their task hour performance over the remainder of the project.
- **Update the task productivity rates** – Team members use the available project to-date productivity rate data to update their team and personal task and schedule plans. Use of actual to-date rates will allow the team to make statistically sound estimates with more accurate prediction intervals for the estimate, to determine the likelihood that their project will fall outside of the prediction intervals, and to calculate confidence levels for various alternate commitment dates that may be needed.

If the overall project commitment, requirements, or resources change significantly, the team should hold a completely new launch with management participation. If the team plan changed so

significantly that a replanning session is insufficient to correct the plan's problems, and the project's requirements, resources, or committed dates are unchanged, the team should conduct a full relaunch.

Knowledge Area 4.5: Phase, Cycle, and Project Postmortems

The postmortem analyzes the collected data on schedule, task hours, estimation, quality, and process. The results are used to provide personal, team and organizational planning data for use in an upcoming launch or relaunch and to identify improvements to process elements. The focus is on using data for planning and process improvement. This knowledge area discusses the postmortem differences, objectives, planning, and processes.

4.5.1 Defining phase, cycle, and project postmortems

There are two types of postmortems that are conducted at the appropriate points in the project lifecycle.

- The *phase or cycle postmortem* is held before any subsequent launch or relaunch and includes only the data on the work completed during the earlier project phases or cycles. Depending on a project's overall duration and needs, the team may choose to use phases, cycles, or both in determining when it needs to conduct a launch or relaunch. A phase represents a part of the development lifecycle such as the implementation phase, and a cycle represents the time between planning horizons. A phase can encompass several cycles, just as a cycle can encompass several phases. The focus of these postmortems is to evaluate interim project status and calibrate planning parameters to revise goals and improve performance in subsequent phases or cycles.
- The *project postmortem* is conducted at the end of the project and includes the full product or project data. Organizational process baseline data may be updated at this time.

4.5.2 Postmortem objective

The objective of any postmortem is to help the team improve quality, design practices, planning, tracking, and teamworking, and to prepare the members to capitalize on this learning when they start another phase, cycle or project. The postmortem also provides the team with an opportunity to gather the available data when it can be most economically, rapidly, and accurately obtained.

4.5.3 Planning for a postmortem

Postmortems should be included as tasks that are planned and estimated when making the long-term and detailed horizon plans.

4.5.4 The postmortem process

The same process is used for conducting phase, cycle, or project postmortems. The only difference is in the scope and timing. The activities performed in a postmortem include the following.

- Baseline evaluation
- Plan evaluation
- Quality performance evaluation

- Process evaluation
- Stakeholder survey
- Goal evaluation
- Postmortem report

More information about preparing for and conducting a postmortem, as well as the content of a postmortem, can be found in Knowledge Area 5.5.

Competency Area 5: Gathering and Using TSP Data

In the same way that PSP relies on individual data, the TSP methods rely on data collection and analysis to enable teams to understand what they do and how they do it, thereby enabling teams to identify effective procedures that should continue to be used and to pinpoint those areas in which improvements are needed. Data provide a baseline for making estimates and plans, tracking progress against the plan, and measuring the effects of changes implemented as part of a process improvement effort. Therefore, it is critical to the successful implementation of TSP that teams collect data as they work, for use in later analyses and estimations.

The knowledge areas composing this competency area are as follows.

5.1 Data Recording – Data are used to provide teams with a baseline for making estimates and plans, tracking progress against the plan, and measuring the effects of changes implemented as part of a process improvement effort. Therefore, it is critical to the successful implementation of TSP that teams collect data accurately and in a timely manner as the project work progresses.

5.2 Gathering and Using Size Data – Size and time are often correlated, and when they are, size estimates can be used to estimate effort and then create plans based on the size and effort estimates. Size data are also useful for tracking development effort and assessing product quality (when defect data are normalized based on size). This knowledge area discusses how the TSP measures size and gathers and uses size data.

5.3 Gathering and Using Schedule Data – Schedule data are used to predict the project's likely completion date and to track actual progress against the planned schedule.

5.4 Gathering and Using Quality Data – The primary focus of the TSP is producing a high-quality product. This knowledge area describes the principles, measures, tools, and techniques for using data to manage quality.

5.5 Gathering and Analyzing Postmortem Data – The postmortem provides the team with an opportunity to learn from its work to improve their product quality, design practices, planning and tracking, and teamwork. By gathering, compiling, and analyzing the available data at the end of a phase, cycle or project when it can most economically, rapidly, and accurately be obtained, the team members can learn from their strengths and weaknesses and capitalize on this learning when they start another phase, cycle or project. This knowledge area discusses the types of data that are gathered and analyzed, identifies the various team member responsibilities for gathering and preparing data for the postmortem, explains how data are used in the postmortem meeting to identify strengths and areas for improvement, and lists the data that should be captured in the postmortem report.

References: The material covered in this competency area is detailed in these primary sources.

[Humphrey 1995, Chapters 4, 5, 6, 7]
[Humphrey 2006a, Chapters 5, 11, 19, 20]
[Humphrey 2006b, Chapter 11]

Knowledge Area 5.1: Data Recording

Data are used to provide teams with a baseline for making estimates and plans, tracking progress against the plan, and measuring the effects of changes implemented as part of a process improvement effort. Therefore, it is critical to the successful implementation of TSP that teams collect data accurately and in a timely manner as the project work progresses.

5.1.1 Collecting high-quality data

The best way to ensure that data is of high quality is to train team members in the proper methods for taking process measures and recording the data that they collect. Using automated data-collection tools can help to improve data quality by providing a convenient means for capturing process information immediately after the data become available. The best way to get high-quality data is to ensure that information is captured in real time (or as soon as possible after the data are generated).

5.1.2 Using data for planning purposes

High-quality data are useful for making accurate plans; however, any data (regardless of quality) is better than no data at all. Whenever possible, every product, job, or project should be planned using estimates that are based on analogous historical data.

- The best estimates are based on actual data from one or more prior products, jobs, or projects of a similar nature.
- The more similar the prior efforts are to the one being planned, the more accurate the estimate is likely to be.
- The more historical data are used when making an estimate, the more accurate the estimate is likely to be.
- Estimating a large job or an entire project as a composite of multiple smaller work products or sub-projects is more accurate than estimating the project as a single large unit.

5.1.3 The TSP planning parameters

TSP plans typically focus on three main parameters: product size, project schedule, and product quality. Historical data (when available) are used to make estimates for the size of each component that will be needed to produce the overall work product and the planned defect injection and removal rates; the size and quality estimates are then used to produce a planned schedule and estimated completion dates for the project phases and overall project. The planning parameters (size, schedule, and quality) are discussed in Knowledge Areas 5.2, 5.3, and 5.4.

Knowledge Area 5.2: Gathering and Using Size Data

Size and time are often correlated, and when they are, size estimates can be used to estimate effort and create plans based on the size and effort estimates. Size data are also useful for tracking development effort and assessing product quality (when defect data are normalized based on size). This knowledge area discusses how the TSP measures size and gathers and uses size data.

5.2.1 Size measures

Size measures quantify how large a work product is, using metrics that are appropriate to the work product. For example, pages (versus words or letters) might be an appropriate measure for text

pages, or lines of code for measuring software components. Size measures apply not only to the final deliverable products, but also to the component parts and interim versions of the product. The TSP uses size measures to define a consistent metric for similar work products and establish a basis for normalizing time and defect data relative to the work product size. The TSP also uses historical size data to help make better size estimates (and therefore, better cost and schedule estimates) for projects of a similar nature.

5.2.2 Physical and logical size measures

Project work components or products can be measured in terms of physical size or logical size.

- A *physical size measure* provides information about the size of a physical entity (the actual number of occurrences of an item in some product). Physical size measures are based on a simple objective standard – the same number is obtained, no matter who is counting. For software projects, physical size measures might be expressed in terms of pages of documentation (for user manuals, test plans, etc.) or numbers of lines of code (LOC) for software components. Physical line of code measures are determined using a defined counting standard that is independent of the programming language being used.
- A *logical size measure* provides size information in terms of groupings of physical entities that can logically be grouped together. The logical size measure does not necessarily correspond to the physical size measure for the same entity, depending on the counting standard defined for the logical measurement. In software, logical size measures for LOC must be determined according to the counting standard appropriate to the programming language being used.

5.2.3 Collecting size data

Size data are most accurate when they are collected in real time, preferably using an automated tool that will record both the planned and actual sizes for the various product parts or components. The tool must calculate totals for each category of size data or otherwise ensure the self-consistency of the data being collected. Among the size data that should be collected are the following parameters.

- Added product size
- Added and modified product size
- Base product size
- Deleted product size
- Modified product size
- New reuse product size
- Reusable product size
- Reused product size
- Total product size

All categories of size data must be self-consistent if they are to be useful for planning and tracking purposes.

5.2.4 Using size data for estimating

As in PSP, the TSP uses a defined estimating process called PROxy-Based Estimating (PROBE) to make size estimates for the components and products to be built. An estimating proxy relates

product size to functions that can easily be visualized. Because a proxy is often easier to visualize than a size measure, proxies can help estimators to judge a product's likely size.

There are four PROBE methods; which method should be used depends on the amount of available historical data. (Please refer to PSP BOK Knowledge Area 3.5 for detailed descriptions of the various PROBE methods, and PSP BOK Knowledge Area 4.2 for guidance on choosing the appropriate PROBE method when making size estimates.) Size estimates for new products are produced during a TSP launch in meeting 4 (see 3.3.5) and are revisited during subsequent phase relaunches, and may be revised as needed. In general, methods C and D are the primary PROBE methods used by TSP teams.

Knowledge Area 5.3: Gathering and Using Schedule Data

Schedule data are used to predict the project's likely completion date and to track actual progress against the planned schedule. (The PSP BOK [Pomeroy-Huff 2009] contains additional information regarding the collection and use of schedule data.)

5.3.1 Collecting schedule data

Schedule data are most accurate when collected using an automated tool that will record planned task names and descriptions, phases in which the work is to be done, product/element involved, applicable committed dates for completing tasks, and the actual dates on which tasks were completed. Schedule data should be collected in real time to the extent possible, particularly information regarding task completion dates, since this is the primary means of obtaining earned value (EV) credit that allows individuals to track their progress against the planned value (PV) for any point in the schedule.

5.3.2 Schedule tracking

At any point in the project, the actual EV earned can be compared to the cumulative PV to determine if the project is on schedule, behind schedule, or ahead of schedule. The various interpretations of EV to PV comparisons are delineated in Appendix C11.

Knowledge Area 5.4: Gathering and Using Quality Data

The primary focus of the TSP is producing a high-quality product. This knowledge area describes the principles, measures, tools, and techniques for using data to manage quality.

5.4.1 Quality measurement principles

The quality of a product or process is a parameter that (unlike time or size) cannot be measured directly; quality measurement must follow an indirect strategy based on an arbitrarily defined standard. A product's quality can only be inferred from the accumulation of measurements on the product's behavior and the processes used to produce, test, or repair it. Typical criteria for such measures include various cost elements, customer satisfaction ratings, regulatory requirements, etc.

Since multiple measures are needed to judge a product's quality, quality measurement is a statistical activity and it must use statistically sound methods. To provide a comprehensive defect profile and to provide statistically useful data, all defect data must be recorded as soon as they are discovered, preferably using an automated tool that will allow each individual defect to be tracked by defect identifier number, defect type, phases in which the defect was injected and removed, defect fix time, a fix reference for fix errors, and a brief description of the defect.

5.4.2 Software quality

Defect management is the principal focus of the TSP, since defect removal efforts largely determine the project cost and schedule performance for software-intensive products and projects. As with PSP, the TSP measures quality in terms of defects. A *defect* is anything in software programs or other products that must be changed for it to be properly designed, developed, maintained, enhanced, or used. Defects can be in the requirements, specifications, designs, code, or other components such as product documentation or user manuals.

5.4.3 The TSP derived quality measures

Derived measures can be either historical or in-process. Historical measures are available only after all process steps providing data for that measure have completed. Historical measures are useful for planning, evaluating a product after a process step, or evaluating the final product at the end of development. In-process measures use data recorded during a process step and can be used to help perform that process step correctly. An in-process measure becomes a historical measure once the process step has been completed.

The historical derived quality measures for TSP include the following.

- *Defect injection rates* – The defect injection rate measure is calculated by dividing the number of defects injected in a phase by the hours spent in that phase – for example, defects injected per hour in requirements, design, coding, etc. Defect injection rates can be used to estimate the numbers of defects in a phase when actual or planned phase times are known. This is helpful when making a quality plan or when estimating defect values when defect data are not available.
- *Defect-removal yield* – The defect-removal yield measure is a measure of the effectiveness of defect removal activities. Yield is calculated by multiplying the number of defects found in a phase (or group of phases) by the number that could have been found, then multiplying that number by 100. The number of defects that could have been removed is the sum of those in the product at phase entry and those injected during the phase. Because the actual number of defects that could have been found will never be precisely known, all defect-removal yield values are estimates.
- *Cost-of-quality (COQ)* – Cost-of-quality (COQ) is a measure of the effort a project or organization expends recovering from defective work. COQ defines quality issues in management and business terms, and includes performance costs (the costs of doing the job in the first place), appraisal costs (the costs of reviewing or inspecting a product and fixing the defects found), failure costs (the costs of compiling, testing, or otherwise automatically examining a product and fixing the defects found), and prevention costs (the costs of devising and implementing measures to prevent failures). COQ measures are used in the

TSP to evaluate the cost-effectiveness of a process, compare processes, and identify improvement goals and opportunities.

- *Process quality index (PQI)* – The process-quality index (PQI) characterizes the quality of a software development process. It measures five elements: design, design-review quality, code-review quality, code quality, and program quality.

In-process derived quality measures used in TSP include the following.

- *Defect density* – The defect density measure is the number of defects injected or found in one or more phases divided by the product size. By normalizing quality in terms of product size, defect density measures enable comparison of the various products and the processes that produced them. Defect density measures are also used to establish quality goals, measure quality performance, and compare processes and products.
- *Defect removal rates* – The defect removal measure is calculated by dividing the number of defects removed in a phase by the hours spent in that phase. Defect-removal rates can be used to estimate the time required to remove the defects in a process phase, establish process quality index parameters, or set personal or team goals for the target times for a defect- removal phase – and then use these goals to guide team members in their work.
- *Phase-time ratio* – Phase-time ratio is the ratio of the time spent in one phase to that spent in another. Example phase-time measures are design time/coding time, design-review time to design time, or code-review time to coding time. Phase-time ratios can be used to establish individual and team quality goals, guide individuals in doing their work, and calculate PQI values.
- *Review/inspection rates* – Review rate refers to the size of product reviewed per hour. This rate is calculated for both reviews and inspections.
- *Capture-recapture estimates* – Capture-recapture estimates provide a way to estimate the likely number of defects remaining after a team inspection. These estimates can be used to guide the team in several ways, such as evaluating the effectiveness of a team inspection or deciding whether to pass a product on to the next phase or conduct some remediation activities.
- *Defect-removal leverage* – Defect-removal leverage (DRL) is a measure of the relative effectiveness of defect removal for any two process phases. For example, the DRL for design review relative to unit test would be defined as DRL(DR/UT) = defects per hour in design review divided by defects per hour in unit test.

The PSP BOK [Pomeroy-Huff 2009] contains more detailed information regarding the derived measures listed above.

5.4.4 TSP quality management methods and tools

Quality products do not happen by accident. Quality must be achieved by the individuals and teams that produce the components and products. The TSP provides a variety of methods and tools to enable high quality at all points in the production process.

- **Reviews** – Personal reviews are conducted by individuals when they examine their own products with the goal of finding and fixing as many defects as possible. Personal reviews should precede any other activity that uses the product (coding, compiling, testing,

inspecting, etc.), and should always precede any inspection. Reviewing before inspection assures inspectors that they are looking for more subtle issues, rather than obvious mistakes.

- **Review checklists** – A review checklist is a tool used in conducting a personal review. Each checklist is specifically tailored to an individual's process. It lists defect categories that have caused problems in the past so that they are checked for during the review.
- **Inspections** – An inspection is a structured review of a product conducted by the product's owner and two or more peers, usually with the aid of an inspection moderator. Inspections follow a defined procedure and have established roles. The object of an inspection is to identify problems in the product. Properly run inspections do not discuss or attempt to solve the problems identified. The TSP inspection process is structured around the following steps.
  - *Planning* – The product owner provides the inspection moderator with the material that will be inspected; the moderator pre-reviews the material to determine if the product is ready for inspection.
  - *Briefing* – At the inspection briefing, the moderator describes the inspection process and the producer reviews the product.
  - *Product review* – The reviewers separately review the product to identify its defects.
  - *Inspection meeting* – The moderator, producer, and reviewers meet to review the product and identify its defects.
  - *Rework* – The producer repairs the product and has the fixes reviewed.
- **Testing** – The role of testing is to exercise programs or parts of programs in a controlled environment and under a defined set of conditions to see if they will produce the anticipated results. The principal purpose of testing is to verify that the product has correctly implemented the requirements with an acceptable level of quality. Testing is essential, but when testing is used to find and fix defects, it is being misused. The fastest and most cost effective ways to remove volumes of defects are with personal reviews and team inspections.
- **Quality reviews** – TSP teams conduct product quality reviews before they release products for testing or other purposes. The objective of the review is to ensure that the team has taken all appropriate steps to ensure a high quality product. Where the process used to produce the product is found inadequate, the team decides what remedial actions to take.
- **Test defect reviews** – TSP teams hold test defect reviews to assess the defects found by testing and to decide on what actions to take to find or prevent similar defects in the future. The data from test defect reviews provide important information about where the development process was deficient. By examining test defect data, teams and team members can identify the types of defects they missed, where the defects were injected, and the steps that did not find them. From these data, teams and team members can both improve their defect-detection activities and define the process actions needed to prevent such defects in the future.

5.4.5 Improving quality management with TSP data

By collecting and analyzing quality data, TSP teams can identify areas for improvement in both individual and team processes. Some of the ways that TSP data can be used to improve quality management include the following.

- **Improve defect-removal methods** – TSP team members can use defect data to improve the cost and yield characteristics of the defect-removal methods. The team members can improve their personal reviews by analyzing the defects that escaped their reviews to identify defect types that should be added to the checklist, to determine why defect types that were on the checklist were not found and make appropriate process adjustments to better find them in the future, and to devise means to better analyze designs to identify and correct all logic defects. Team inspection data can be analyzed to identify the defects that escaped inspection and make additions to the team inspection checklist, to review the team's inspection practices to see where and how to improve inspection yield, and to use the capture-recapture data to adjust the team's criteria for reinspection.
- **Improve the inspection process** – TSP teams can use PQI and defect data to assess the criteria for product reinspection, rework, or replacement.
- **Defect prevention** – TSP teams can use defect data to improve the defect-prevention process, by choosing and focusing on preventing specific classes of defects, devising prevention strategies and needed process changes, testing and evaluating the process changes, and adjusting the process for release into general use.
- **Adjust the legacy-product improvement process** – Teams can use system-test and customer defect-report data to improve the quality of legacy products. Data can help the team to consider the criteria for deciding when to rework components and repair them before modification and when to replace components.

Knowledge Area 5.5: Gathering and Analyzing Postmortem Data

The postmortem provides the team with an opportunity to learn from its work to improve their product quality, design practices, planning and tracking, and teamwork. By gathering, compiling, and analyzing the available data at the end of a phase, cycle or project when it can most economically, rapidly, and accurately be obtained, the team members can learn from their strengths and weaknesses and capitalize on this learning when they start another phase, cycle or project. This knowledge area discusses the types of data that are gathered and analyzed for the postmortem, identifies the various team member responsibilities for gathering and preparing data for the postmortem, explains how data are used in the postmortem meeting to identify strengths and areas for improvement, and lists the data that should be captured in the postmortem report.

5.5.1 Gathering data for the postmortem

When gathering data, it is generally wise to review the reasons for which the data are needed and how they will be used. The critical data and analyses needed for an effective phase, cycle, or project postmortem are the following.

- Goal tracking data to evaluate and improve goal planning and achievement
- Plan versus actual performance (task growth, task hour history, requirements dynamics) to assess and improve process performance
- Estimating accuracy data (productivity and size data) are analyzed to allow improvements in estimating accuracy
- Quality performance data (yield data, defect injection and removal rate data, quality profile data, test data) are analyzed to assess and improve quality performance

The postmortem should be held as soon as possible after the work is completed at the end of a phase, cycle, or project so that data can be gathered when they are most readily accessible and are most likely to be accurate. If the data are not collected when readily available, the team is likely to have trouble reconstructing it later.

Six team roles are responsible in part for ensuring that the postmortems are promptly held and effectively conducted: the team coach, the team leader, the process manager, the planning manager, the quality manager, and all team members.

5.5.2 Preparing for the postmortem

To make the postmortem most useful, the planning manager, quality manager, and process manager must prepare for it before holding the meeting. Advance preparation provides the team with the data needed to assess their performance or to obtain useful improvement guidance.

- The process manager should ask the team members to think of ways to improve the team's processes and to submit process improvement proposals (PIPs) with their improvement ideas, which will be reviewed and discussed at the postmortem. The team members should identify the kinds of data they would like to discuss, then gather these data and make them available to the process manager in time for the data analysis to be completed before the meeting.
- The planning manager should consolidate and analyze the team's actual versus plan data for size, resource, and schedule estimates.
- The quality manager analyzes the quality of the products produced, and the team's performance versus the goals and quality plans.
- The team leader should ensure that the role managers adequately prepare for each postmortem and that the team members provide data and other input as required. After the postmortem, the team leader should ensure that the PIPs identified in the postmortem are promptly and properly handled. At the end of the project, the team leader should guide the team in using the postmortem data to produce the final project report.
- Every team member should participate in the team postmortem by providing data, PIP suggestions, and other requested information. The team members should each provide pertinent data, comments, and suggestions on how to better perform the duties of the role manager positions that they filled prior to the postmortem.

5.5.3 Conducting the postmortem

The postmortem process consists of a number of steps that can be performed in almost any convenient order. These steps are as follows.

- **Baseline evaluation** – How did the configuration management system work and how could it be improved?
- **Plan evaluation** – How accurate was the plan and how could the plan and planning process be improved?
- **Quality performance** – How effective was the team's quality management process and how can it be improved?
- **Process evaluation** – What PIPs can team members suggest to improve any of the team's processes for the next phase, cycle, or project?

- **Stakeholder survey** – Where stakeholder feedback can be obtained, what guidance can the team obtain for its process improvement plans?
- **Goal evaluation** – The team evaluates its performance against its goals.
- **Teamwork assessment** – The team reviews how well it worked as a team and makes suggestions on how teamworking could be improved.
- **Leadership assessment** – The team leader asks the team for comments and suggestions on how he or she could better guide and support the team.
- **Coaching assessment** – The coach asks the team for comments and suggestions on how he or she could better support the team.
- **Postmortem report** – The team decides on the content of the postmortem report, produces a preliminary report plan, and assigns the report-preparation work to team members.

5.5.4 Preparing the final project report

At the end of a project, the team leader or another member of the team gathers all of the meeting data, checks to ensure that they have complete data on the project, and produces the final postmortem report. The report is used to record key data for every TSP project. The final project report is needed by teams and their organization to estimate and plan subsequent projects, to evaluate project performance against plans, to establish future goals, to improve process performance, and to demonstrate process adherence. The report should include the following information.

- Project name (with release, version, or other number as needed)
- TSP team leader(s) name(s), TSP team coach(es) name(s), and number of team members
- Initial project launch date
- Development environment and programming language
- Management systems and support tools
- Code complete date
- System and acceptance tests with start and end dates
- Total task hours by process phases
- Total defects by process phases
- Size for all products created and modified
- Process performance summaries for each unique process or sub-process used
- All other information used as part of the postmortem analysis
- Summary of all conclusions made from the postmortem analysis
- List of improvement actions to be taken

Competency Area 6: Scaling Up the TSP

Successful introduction of TSP into an organization begins with one or two small-scale pilot projects. Pilot projects provide the management support and compelling evidence needed to convince the organization's general population that the TSP methods are effective and will be beneficial to the organization. If the pilot projects are successful, management often decides to implement TSP in one or more divisions or organizations. As with the introduction of the pilot projects, special considerations must be addressed to increase the likelihood that organization-wide implementation will be successful.

Whether or not organizations choose to implement TSP throughout the company, they all have unique needs that may require some tailoring of the TSP applications. This is particularly true if the TSP project team requires more 15 to 20 members, if the members of the team have different professional capabilities or specialties that must work together to produce the product, or if some of the team members work at locations apart from most of the team.

The knowledge areas in this competency area describe the activities of scaling up the TSP for entire organizations or very large TSP project teams, and the adaptations to the basic TSP process that may be needed to address the needs of specialized TSP teams.

6.1 Organizational Implementation – This knowledge area describes the process of scaling the TSP implementation up from use on a few pilot projects to full introduction of TSP across the organization.

6.2 TSP Process Variations – In development work, teams typically are classified as either project teams or functional teams. A *project team* is one that is formed to accomplish a specific project objective and, when that objective has been completed, the team is either disbanded or given another project assignment. A *functional team* is one that has a continuing mission responsibility. Additional variations in team type are due to team size or physical location. TSP can be adapted to fit the needs of functional teams (TSPf), integrated project teams (TSPI), distributed teams (TSPd), multiple TSP teams working in tandem (TSPm), TSP teams that also use CMMI (TSP+), and academic (student) teams (TSPi). This knowledge area describes the TSPf, TSPI, TSPd, TSPm, TSP+, and TSPi process variations.

6.3 Large-scale TSP Teams – This knowledge area describes the characteristics and considerations unique to large-scale TSP teams.

References: The material covered in this competency area is detailed in these primary sources.

[Humphrey 2000]
Humphrey 2006a, Chapters 21, 22, 23, 24]
[Humphrey 2006c]

Knowledge Area 6.1: Organizational Implementation

This knowledge area describes the process of scaling the TSP implementation up from use on a few pilot projects to full introduction of TSP across the organization.

6.1.1 Planning for organization-wide introduction

The overall objective of the scale-up effort is to broaden the use of the TSP to an entire organization. As soon as the pilot projects have demonstrated the benefits of the TSP to management's satisfaction, the organization should start planning for full TSP introduction. Top-level management should publicize the results of the pilot projects, using testimonials for these projects wherever appropriate. Management should also appoint an implementation team, including a TSP coach, to begin planning the implementation effort. This team should act quickly to carry out the following tasks.

- Obtain budget information or planned target costs for the scale-up effort
- Ask management to define its quality, productivity, or predictability targets for the effort
- Make a long-term TSP introduction plan, and have it reviewed and approved by the appropriate managers and executives
- Identify the management teams that are most supportive of the TSP and target their groups for the next projects
- Start efforts to recruit and train internal coaches

6.1.2 The scaling-up strategy

The scaling-up strategy should start with a single organizational location; ideally, all of the pilot projects should have been implemented at this location. This helps to achieve the strategic objective, which is rapid building of an environment in which most or all of the work is being done by TSP teams. Once this objective is close to being achieved with the first location, start on subsequent locations in the organization. When the organization has built a sufficiently capable internal pool of TSP coaches, it might be possible to initiate scale-up efforts in several locations at the same time.

6.1.3 Requirements for successful scaling-up efforts

Every new location should be treated as though it were an initial pilot project effort. This means that, ideally, all of the teams to be piloted in a location or division come from the same location or division. However, if the organization managers so desire, the scale-up process can be accelerated by assigning members from the next target location to work on teams at an existing TSP organization. These assignments can be particularly helpful for accelerating the growth and development of new team leaders and coaches.

Any successful TSP introduction requires support from qualified TSP coaches [Chick 2009]. Therefore, when deciding the order in which TSP will be introduced at various locations, the organization should consider the availability of external and internal coaching support for each location. Qualified coaches may be available from local process improvement organizations or from a local university. If such support is not currently available, it may be possible for an organization to convince such groups to build a TSP support capability and to offer the capability in support of the scale-up effort.

6.1.4 Training requirements for TSP scaling-up efforts

As with initial pilot introductions, successful TSP implementation requires that all team members be properly trained in the TSP and supporting methodologies. Ideally, the team members should be fully trained prior to the organizational roll-out.

- Before the initial team launches at a new location, provide all engineer/developer team members with the full PSP training and other team members with the introductory PSP and TSP training. All team members should have basic training in the TSP support tool to be used by the team. This training can be followed by advanced TSP courses, if needed.
- The advantages of this strategy are that the training is most likely to be completed in time for the team members to be fully effective on their teams, and a high percentage of these teams are likely to have successful projects.
- The disadvantages are that the training is long and may preclude in-depth coverage of the content which could increase resistance to TSP introduction in locations where team members are skeptical of the effort. Also, since much of the training is provided well in advance of the project kick-off, team members may not retain as much of the information as they will need for optimal participation in the initial introduction efforts.

However, in extreme cases, the teams can receive just-in-time TSP training. All team members should be provided with basic PSP training or introductory PSP/TSP training, as well as the TSP tool training needed to participate in a team launch. Immediately after the launch, team members should receive the additional training required for team working. More advanced PSP and TSP training could then be provided if needed.

- This introduction strategy is practical after an organization has gained enough TSP experience to have proof of the method's effectiveness, or when introducing a one or two new members into a team that already is successfully and competently using the TSP.
- The advantages of this strategy are that the initial training duration is limited, making TSP introduction more attractive and improving retention. Also, since teams would expect further training, it likely makes it somewhat easier to schedule short advanced courses as needed.
- The disadvantages are that some or all of the training needed for team working might not be provided once the team starts the project. Further, without the full PSP training, team members might not have the motivation needed to follow the PSP planning and quality management methods, thus making it more likely that these partially-trained teams would fail or have marginal results.

Knowledge Area 6.2: TSP Process Variations

In development work, teams typically are classified as either project teams or functional teams. A *project team* is one that is formed to accomplish a specific project objective and, when that objective has been completed, the team is either disbanded or given another project assignment.

A *functional team* is one that has a continuing mission responsibility. Additional variations in team type are due to team size or physical location. TSP can be adapted to fit the needs of functional teams (TSPf), integrated project teams (TSPI), distributed teams (TSPd), multiple TSP teams working in tandem (TSPm), TSP teams that also use CMMI (TSP+), and academic (student) teams (TSPi). This knowledge area describes the TSPf, TSPI, TSPd, TSPm, TSP+, and TSPi process variations.

6.2.1 Types of TSP teams

The TSP team is typically a group of 3 to about 15 members that has a common plan, defined goals, and a single team leader. However, there are six common team variations to which TSP can be applied.

- On *functional teams*, all of the members do similar work, but their individual tasks are usually independent of each other. Although several members may occasionally work on elements of a common product release, they usually work alone.
- *Integrated project teams* have members with different specialties, such as hardware design, systems integration, software implementation, and test, and all of the various specialists must work closely together effectively to produce a quality product.
- On a *distributed team*, the members work in different physical locations. Distributed teams may be either project or functional teams, and either stand-alone or multiple teams.
- The *TSP multi-team* structure is composed of two or more teams working in tandem on different element of the same products, and is generally used on projects that are larger than a single team could handle by itself.
- For *TSP teams working in an organization that uses CMMI* (Capability Maturity Model Integration) to guide process improvement efforts, TSP+ extends TSP and TSPm by modifying or adding specific process elements that explicitly address certain aspects of CMMI, including additions for launching and working as a process group. TSP+ is a primary component of the *Accelerated Improvement Method (AIM)* which uses TSP as the primary implementation vehicle for CMMI practices at maturity levels 2 and 3.
- The *academic (student) TSP team* is composed of three to five individuals who are enrolled in an undergraduate or graduate course of studies in computer science or software engineering. The introductory TSP methodology builds on the pre-requisite PSP training and gives students the needed conceptual understanding and basic skills needed to work on TSP teams after they have completed their studies. However, the TSPi does not contain all of the material needed for successful implementation of the TSP in a commercial or government project setting and should not be used as a substitute for the complete TSP methodology or the TSPf, TSPI, TSPd, or TSPm process variations.

6.2.2 The TSP functional team (TSPf)

A functional team is one that has a continuing mission, rather than a temporary project objective. There are three typical types of functional teams.

1. **Resource pools** – These groups are used in matrix organizations to house the individuals who are assigned as needed to project teams. They use whatever process is appropriate to the project team work to which they are assigned.
2. **Capability groups** – These groups provide a centralized skill set to projects or other users. Examples are application development groups, maintenance groups, quality assurance staff, and process groups.
3. **Operational groups** – These groups have continuing responsibility for some ongoing operation like maintaining and enhancing a product line, operating a model shop, or managing a complete organization.

The capability and operational groups have similar needs for a functional-team process.

The principle challenge with a functional team is that the members may not act and feel like a team. Because the team members have relatively independent tasks, they do not have the integrating focus of a common product or mission. However, the team can develop a cohesive spirit and energy, if they follow the general TSP launch strategy of working together to develop its goals, strategies, processes, and plans. The team members should work as a unit throughout the launch, except in meeting 6 when members make their individual plans. Although working as a single team will take more time than having each team member do a separate mini-launch, the team approach shows the team members how to do a launch and facilitates teambuilding. When functional teams stay together for the entire launch, the entire team participates in planning each member's work, thereby gaining an appreciation for what the other team members are doing. The team members learn about and can capitalize on the other members' skills and knowledge, and may discover opportunities for cooperative work. They begin to feel more like a cohesive team with a common purpose.

6.2.3 The TSP integrated project team (TSPI)

An integrated project team is one that is formed to accomplish a specific project objective and, when that objective has been completed, the team is either disbanded or given another project assignment. The team may be of any size from three members up to hundreds or thousands, and includes everyone who has a full-time project assignment and whose skills are required to perform the project. The team is typically organized around the end product and may have many sub-product groups, but the work of the team as a whole is highly interdependent and focused on the single end product. Integrated teams should be kept to the same size as with regular TSP teams, that is, three to fifteen members. The team should be limited to members from the groups that will actively work on the project during the immediate next phase. The team may include hardware and software engineers, test engineers, requirements or systems people, application specialists, and even customer representatives. In very large projects the teams may be organized by function such as a systems engineering team. These function based teams can still use TSP, with some basic modifications to the regular TSP scripts, forms, and measures which account for the non-developer activities [Carleton 2010].

Because integrated product teams usually include one or more non-software components, not everyone on a TSPI team is a software developer. The non-developers on the team will have to be trained in the vocabulary and conceptual framework of the PSP and TSP by completing the *TSP Team Member Training* course or similar instruction. The scripts, forms, measures, processes and other project parameters will have to be tailored to fit the needs of the various specialized functions of the various team members, while maintaining sufficient commonality that the team as a whole can plan and track progress against cost, schedule, and quality goals using a consolidated project plan that is based on traditional project planning and tracking methods [Chick 2006].

6.2.4 The TSP distributed team (TSPd)

A distributed team is composed of members who are in different physical locations. Because of the complexity of modern systems and the distributed nature of the modern workforce, it may be impractical to locate complete teams in one facility. Members of a distributed team may be evenly divided between two or more locations, have most of the team members in the same location with

one or two members working remotely, or have all of its members scattered across various locations.

To sustain team energy and motivation, team members need to maintain good communication among the members and receive regular information on project status, team leader and management decisions, task assignments, and the team leader's and management's evaluation of their personal performance. The team leader of a distributed team must pay particular attention to the communication needs of each team member.

In view of the potential communication problems, the role manager assignments are particularly important on distributed teams. Assigned roles allow all team members to know who has responsibility for a particular team function, and therefore knows who to contact when particular needs arise. Because each role managers has the responsibility for coordinating role activities with other team members as needed, the assignment of role manager positions to members of distributed teams also ensures that all team members are informed and involved in facets of team management that requires broad team participation. For this reason, all team members including those at remote locations should have role assignments to keep them in communication and involved with the other team members.

Team communication problems can also be lessened or alleviated by ensuring that the team holds frequent meetings, even if only by telephone or videoconference. During these meetings, the team members should review the work accomplished since the past meeting and their plans for the next work, discuss any issues or problems and seek help in resolving them, and track the open issues each week until they are resolved.

6.2.5 The TSP multi-team (TSPm)

As the scale of modern technological products increases, the sizes of their engineering teams also grow. TSP teams with more than 15 members should be divided into one or more smaller teams. When two or more TSP teams are working in tandem to meet the same project goals and produce elements of the same product and the team leaders are members of one leadership team, the result is called a *multi-team*. (If projects are so large that it is impractical to have all team leaders be members of the leadership team, the team is not a multi-team, but rather a large-scale team, as discussed in Knowledge Area 6.3). The leadership team provides overall management coordination and guidance for the multi-team. The leadership team consists of the project manager and the leaders of all of the teams. There are also role manager teams, each composed of the same type of team role manager. The function of the role manager teams is to coordinate the activities of the teams in the areas covered by the roles. The role manager teams should meet weekly to identify and resolve issues and to address topics that the leadership team delegates to them.

The individual teams that compose a multi-team are regular TSP teams, each with 15 or fewer members and its own plans, defined goals, team member roles, and team leaders who guide the work of their own teams. Because each TSPm team is a complete team, each requires coaching and needs a separate coach during the TSPm launch. When two different specialty teams are producing different but closely related products, as in the implementation of the hardware and

software parts of a system, each type of specialty should be on its own separate team. However, if two different specialties – such as systems and software groups – are producing the same product (such as a specification document), these groups should be on the same team, as long as it does not make the team too large.

6.2.6 The TSPm launch

Each TSPm multi-team is built with the TSPm launch process. A TSPm launch normally takes five very full days, rather than the two to four days for a typical single-team TSP launch. The additional time is required for leadership and role manager meetings and to resolve cross-team issues. The standard TSP launch meetings are followed by evening leadership team meetings with the individual team coaches to review the launch status, settle outstanding issues, and provide any needed guidance to the team leaders and coaches.

The TSPm launch is complicated by large group size and the need to coordinate the actions of the several teams. Therefore, the TSPm launch contains several supplemental meetings in addition to the regular launch meetings.

- **TSPm launch meeting 1A –** The project manager meets with the entire team to review the project strategy with all project team members, to divide the overall team into sub-teams of about 3 to 15 members, to provide these sub-teams with sufficient guidance so they can plan their portions of the overall project, to establish mechanisms for these sub-teams to coordinate their work and adjust their responsibilities as they learn more about the project, to allow the design team to describe the overall product conceptual design, and to summarize the responsibilities that the leadership team has assigned to each role manager team.
- **Launch meetings 2 and 3** are conducted separately by each sub-team to set its own goals, select role managers for each role, and produce the conceptual design and other meeting 3 artifacts for their portions of the project.
- **Launch meeting 3A –** The leadership team meets to review the launch status, identify and resolve issues from meetings 2 and 3, review the team role assignments, and designate the leader for each role manager team.
- **Launch meetings 4 and 5** are conducted independently in each sub-team to produce the overall plan and quality plan for their portions of the project work.
- **Launch meeting 5A –** The leadership team meets to review the overall launch status, identify and resolve issues from launch meetings 4 and 5, and review summary reports from the planning manager role team and quality manager role team.
- **Launch meeting 5B –** The planning manager role team meets to consolidate the sub-team plans to produce the overall team plan, to identify internal and external team dependencies, and to prepare a summary report for the leadership team.
- **Launch meeting 5C –** The quality manager role team meets to review each team's quality plan, to consolidate the sub-team quality plans to generate the overall quality plan, and to prepare a summary report for the leadership team.
- **Launch meeting 6** is held separately for each sub-team to allow team members to produce their individual plans and for the sub-teams to consolidate their next-phase plans with a balances workload.

- **Launch meeting 6A –** The leadership team meets to review the overall launch status, review the planning manager role team's summary report, and to identify and resolve issues arising from meeting 6.
- **Launch meeting 6B** – The planning manager role team meets to review the detailed plans of each sub-team and consolidate them to produce the team plan, identify any planning issues that might need to be addresses across the sub-teams, and prepare a summary report for the leadership team.
- **Launch meeting 7** is conducted by each sub-team to identify potential risks and produce a risk mitigation plan for high-priority risks.
- **Launch meeting 7A** – The leadership team meets to review the overall launch status, review each sub-team's risk assessment, identify and resolve any remaining issues, and plan for the management meeting.
- **Launch meeting 8 –** Each sub-team prepares for and out-briefs their plan to the leadership team and other sub-teams. Then the leadership team consolidate the sub-team out-briefs into a single out-brief for the project manager and management team.
- **Launch meeting 9** – The leadership team presents the team plan to the project manager and management team.

6.2.7 TSP+

TSP+ is an extension of TSP and TSPm that has been developed to extend TSP support to provide additional performance benefits in process areas that have more of an organizational focus. Among these extensions are clusters of process elements (scripts, forms, guidelines and specifications) that support practices in configuration management, measurement and analysis, quality assurance, organizational training and decision analysis for development teams, and specific extensions for process groups to use TSP methods to establish and maintain the organization's standard process assets and data. These extensions also increase TSP's coverage of the CMMI-DEV at maturity levels 2 and 3.

6.2.8 Introductory TSP for academic (student) teams (TSPi)

Whereas TSPI is a specific designation for an integrated project team that is using the TSP on a large project, TSPi is an introductory variation of TSP that was designed for use by small teams on short-term projects as part of a course or curriculum in software engineering in an academic institution. The introductory TSP methodology builds on the pre-requisite PSP training and gives students some experience in applying the concepts and basic skills that they will need in order to work on TSP teams after they have completed their studies. However, the TSPi does not contain all of the material needed for successful implementation of the TSP in a commercial or government project setting. TSPi contains only five standard roles and has different scripts, standards, specifications, and forms than those used in TSP. In addition, the TSPi has been adapted for use in a semester-long implementation, and is designed to be implemented through a series of pre-determined exercises that simulate, but cannot replicate, actual project experiences. Therefore, the TSPi should be used only as part of an academic course of study and should not be used as a substitute for the complete TSP methodology or the TSPf, TSPI, TSPd, or TSPm process variations.

Knowledge Area 6.3: Large-scale TSP Teams

This knowledge area describes the characteristics and considerations unique to large-scale TSP teams.

6.3.1 The large-scale TSP team

A TSP multi-team is considered a large-scale team when it becomes too large for all team leaders to be members of the leadership team. Leadership teams larger than the typical 15 to 20 member limit for teams are rarely able to develop the degree of cohesion desired in a leadership team and necessary to establish and maintain the multi-team's common goals, processes, and plans. A large-scale team typically is managed by a program manager, with a program management staff and multiple leadership teams leading TSPm teams that work on the major sub-systems or components of the overall program.

6.3.2 Processes and practices for large-scale TSP teams

Processes for large program-wide teams must be customized or tailored to meet the precise needs of the development program, using whatever process elements are available and have been proven effective on prior projects. An overall coordinating management process must also be customized. Unless the team managers and their teams consistently follow sound processes and practices, no larger-scale process can be effective. Whatever processes are chosen, they should be congruent with the standard TSP principles.

- Teams should be given aggressive development goals.
- Teams must make their own development plans and track their progress against them.
- Teams must negotiate their commitments with management.
- Team must measure and manage the quality of their work.
- Teams and team members must manage themselves and their workloads.
- Every team member should use sound engineering methods.

Large-scale teams have some unique issues that require the TSP sub-teams to conform to the same set of standards, procedures, and measures, rather than each team using unique metrics. This facilitates the coordination of information across numerous teams, and enables program-wide workload balancing in the event that one or more teams encounter issues which threaten their schedules or quality plans. Therefore, all large-scale projects must establish the following.

- Standard and consistent project management standards and reporting measures
- Coordinated program milestones and defined commitments
- Mechanisms for documenting all decisions made at the team level and making them available to the entire program to allow coordination across teams and management levels
- Architectural standards and a set of architectural resolution and control procedures
- Processes for defining, maintaining, measuring, evaluating, and controlling the system's emergent properties
- Mechanisms for monitoring the quality of every system component produced and taking corrective action whenever needed
- A project-initiation team composed of a small core of experts who define the program concept and requirements, establish the program goals and strategy, and convince management to sponsor and initiate the development effort

- An organizational structure to address large-system program management issues such as responsibilities of managers at various levels, handling program-wide communications, and making program-wide decisions

6.3.3 Large team communications

When a large team is composed of multiple work groups or teams, the members tend to focus on their personal work and leave external team communication to the team leader, often leading to mistakes, oversights, and misunderstandings. On large-scale teams, role managers of individual teams should consider discussing problems with their counterparts on other teams, since communications among role managers can often resolve cross team issues at the level where they are best understood. Other problems may need to be communicated by team leaders up the management chain to the appropriate levels, using the program-wide communication strategy.

Appendix A: Engineering Guidelines

A1: Principles of Modern Engineering Work

A1.1 Develop useful and economical products

The development engineer's primary job is to produce high-quality work products that can be predictably produced, distributed, and used at affordable costs.

A1.2 Use scientific methods

Qualified engineers must understand the scientific methods and practices that are relevant to their work and use these methods and practices on the job.

A1.3 Commit to excellence

Engineers must consistently strive for excellence, and recognize that it is faster and cheaper to do the job right the first time.

A1.4 Be persistent

Development involves new problems and challenges, and engineers must be flexible in their approach to their work in order to successfully overcome those problems and challenges.

- Engineers must be willing to learn from their failures. Failures provide insight into how to do the work differently and better. Failures also help engineers to recognize that some things simply cannot be done, and provide the experience needed to know when one approach should be abandoned in favor of a different solution.
- Engineers must learn to consult with their colleagues, rather than trying to do their work alone. Peers and managers often have useful ideas or insights into similar problems and can provide helpful guidance towards generating solutions.
- Engineers must always plan their work, and be willing to change the plan when necessary. The nature of development work means that the plan will always change. Every plan change should be carefully planned and tracked, since unplanned changes tend to require more effort than indicated by a casual estimate.

A1.5 Meet business needs

Engineers should strive to make each project successful from both an engineering and a business perspective. Successful project execution requires both individuals and groups to work with the project's management and customers in order to ensure that the following goals are met.

- Realistic plans and schedules are established
- The work is reprioritized as needed
- Projects are properly staffed and team resources are rebalanced regularly
- All changes are planned and managed
- All parties maintain a focus on quality

A1.6 Modularize

A fundamental principle of engineering work is modularization.

- Start with an overall product concept and subdivide it into multiple smaller elements.
- Define the characteristics and interfaces for each element.

- Continue subdividing and defining each component until the elements are of a suitable size for development by a small team or individual.

A1.7 Design

To properly plan a project, engineers must produce designs for each product level, starting with requirements, and continuing through implementation and test planning. Designs should be documented using precise and clear notation, and should be reviewed and inspected to ensure that each design is of the highest possible quality. As the work progresses and the design evolves, the design documents should be updated to reflect the most current iteration.

A.1.8 Maintain product focus

A successful project requires engineers to first define measures of success for the work they are going to perform. Once the criteria for a successful project have been established, they should be used to establish short-term goals that enable individuals or teams to maintain their priorities. As the work progresses, the engineers should strive to overcome all obstacles, seeking advice from peers and managers or even changing direction when necessary. Engineers should also involve the customer wherever possible to ensure that the product meets the customers' needs.

A1.9 Recognize requirements uncertainty

Engineers must recognize that requirements are almost never complete or accurate. Requirements cannot be known before the product is completed and used, and the initial requirements will change as more is learned about the product and its use. Large requirements changes rarely cause problems because the scale of these changes requires engineers to estimate and plan the work that will be needed to implement the new requirements. Problems usually arise because of the cumulative effect of many small changes in requirements that are not adequately planned. To minimize the impact of requirements changes, engineers must produce detailed plans and work with the customer to resolve their requirements assumptions as the work proceeds. Changes should not be implemented until all parties are fully aware of the impact of those changes and the resultant effects on budget and schedule.

A1.10 Meet professional obligations

All engineers have a professional obligation to contribute to their profession by

- using the available standards in their field wherever possible
- making others aware of inadequate or incomplete standards, and participate in making improvements
- publishing their significant failures and results so that others can learn from them

A1.11 Plan for future challenges

Modern engineering work is often challenging, and future projects will be even more challenging. Engineers should view each project as an opportunity to improve their skills for the more challenging work to come.

Appendix B: Project Management Guidelines

B1: Operational Processes for Project Management

B1.1 Operational processes

An *operational process* defines precisely how to do a job or task. An operational process provides enough detail to guide a knowledgeable professional in doing that job by providing a framework for making detailed plans with defined tasks, taking process measurements, enabling process analysis and evaluation, and facilitating process improvement.

B1.2 Operational process requirements

Operational processes must

- be clear and concise
- contain only the information that knowledgeable users need to enact the process
- specify the process data to be gathered
- describe when the process data are to be gathered, used, and analyzed
- define the process steps with sufficient precision to enable detailed project planning and tracking

B1.3 Operational process definition

There are eight steps in defining an operational process.

1. Determine needs and priorities.
2. Define process objectives, goals, and quality criteria.
3. Characterize the current process.
4. Characterize the target process.
5. Establish a process development strategy.
6. Define the initial process.
7. Validate the initial process.
8. Enhance the process.

B.1.4 Operational process customization

When an existing operational process is applied to a new situation, it is often necessary to tailor or customize the process to address the particular project situation, since a process that works well in one environment may not be effective in another. For example, a prototype process may not require team inspections; however, if the finished prototype will become part of the finished product, the process should be modified to include inspections.

B2: Project Management Using TSP

B2.1 TSP plans

A complete TSP project plan consists of the baseline (or committed) plan, the team plan, detailed personal plans, and the quality plan.

- The *baseline plan* is the plan to which the team and management mutually agree during launch meeting 9; it can also be called the committed plan because the team made a commitment to management to fulfill the details in the plan. The baseline plan defines the team's major deliverables and the schedule milestone dates. Unless the plan is revised and the revisions are approved by management, the baseline plan provides the benchmark against which management will measure the team leader's and team members' performance.
- The *team plan* guides the team members in their work. Initially, the team plan is identical to the baseline plan, with both containing the same task lists and committed milestone dates. As the work progresses, the team members will add unanticipated tasks to their plans and delete tasks that are deemed unnecessary; therefore, during the execution of the plan, the team and baseline plans gradually diverge. The principle challenge for TSP teams is to ensure that even if the tasks change, the team plan is still able to meet the committed products and dates established in the baseline plan.
- Each team member has a *detailed personal plan* to guide their work for the next few weeks or months. As the members begin to work, they learn more about the project, and their plans begin to change so as to continue to accurately represent the evolved understanding of the necessary individual tasks. Team members must regularly track their progress to ensure that their updated personal and team plans continue to meet the milestone commitments made in the baseline plan.
- The *quality plan* addresses the defect issues associated with the product. The members agree on measurable quality goals, the actions that they will take to achieve those goals, the quality commitments that they will make to management, and the team members' responsibilities for managing and tracking these quality commitments.

B2.2 Updating or replacing TSP plans

Replanning is an integral and normal part of the Team Software Process. TSP plans should be periodically reviewed and updated, or, in the case of drastic changes to project resources or requirements, replaced with new plans. There are six principle reasons for updating TSP plans.

1. **The team made a poor estimate**. Even when TSP teams use historical data, they may occasionally seriously over- or under-estimate one or more parameters of the project. This commonly happens when the project contains unfamiliar tasks that were erroneously thought to be much simpler or substantially harder than was actually the case.
2. **The team's understanding of the work has changed.** As team members perform the tasks on their plan, they learn progressively more about the nature of the project work. New or evolving understanding of the work sometimes affects the project size or scope to such a degree that a new plan is required.
3. **The resources available to do the work have changed.** Because of other demands on their time, team members have over- or under-estimated their available task hours during planning. Management may reassign some team members on a part-time or full-time basis to other projects, or may assign new team members to the project. Any of these factors will affect the schedule and task allocation, requiring the plan to be updated or completely redone.
4. **The project requirements have changed.** Any change, no matter how seemingly small, can affect the schedule and cost of the project. The team should always assess the potential effect on the baseline plan for each requested change and agree to incorporate the change

only if management and/or the customer agrees to the necessary schedule and resource adjustments.

5. **Management's or the customer's priorities have changed**. As with a change in project requirements, the team should always assess the effect of the change in priorities and inform management and/or the customer of the cost the change.
6. **The team has completed or nearly completed its currently planned work.** Detailed plans are accurate only for a few weeks or months at a time. At the end of a project phase or cycle, the team must replan the work for the coming phase or cycle, adjusting their estimates (as needed) based on the team's to-date historical data for the project.

B3: Managing TSP Plans

B3.1 Improving size estimation accuracy

Individuals or teams can improve their size estimation accuracy by

- reviewing estimation errors, identifying their causes, and adjusting the estimation process to address and correct the causes
- reviewing the suitability of the proxies used and examining alternative proxies
- updating team and team-member proxy data

B3.2 Improving task time estimation accuracy

Individuals or teams can improve their task time estimation accuracy by

- reviewing task time estimating errors and determining their causes
- devising means for correcting prior estimation errors, such as using the most relevant historical data or asking for data from similar projects carried out by other TSP teams
- improving the management of non-project demands on team member time

B3.3 Improving schedule estimation accuracy

Individuals or teams can improve their schedule estimation accuracy by

- reviewing schedule estimation errors, identifying their causes, and adjusting the estimation process to address and correct the identified causes
- ensuring that the data used for estimating are current and relevant to the project work
- improving the team's task time (see B3.5)

B3.4 Improving project-completion estimation accuracy

Individuals or teams can improve the accuracy of their project completion estimates by

- determining the causes of past estimation errors and adjusting the estimation process to address and correct the identified causes
- ensuring that tasks are decomposed to the appropriate level of granularity
- improving control over time on-task
- using appropriate data and methods to estimate project completion dates

B3.5 Improving team task time

The fastest and most effective way to improve schedule performance and reduce project costs is to improve the team's weekly task time. There are three general approaches for improving the team's time on-task.

- **Optimize team support.** The team leader, the team coach, or higher management may arrange for improved computing and administrative support, arrange for needed team-member training, obtain expert consultation for optimizing use of available tools and methods, or recommend adoption of better tools and methods than are currently being used.
- **Optimize the team's working environment.** For co-located teams, this may require provision of private team member workspace when needed, and ensuring availability of group workspace as necessary. For distributed teams, this may require ensuring an effective configuration management environment which is directly accessible by all team members, providing a distributed test environment, and ensure availability of effective communication tools (such as video conferencing, web conferencing, message boards, and secure email) for both peer-to-peer and team communications.
- **Limit team and team member interruptions.** The team leader, team coach, or higher management may need to take steps to control the amount of noise and distraction in the team workspace, control meeting frequency, establish uninterruptible team and team member quiet times, or obtain management agreement to limit demands on team member time.

B4: Managing Team Communication

B4.1 The need for team communication

Real communication is more than just the transmission or reception of information. Real communication requires two-way interaction so that both parties can reach a mutual understanding and agreement. Communication among team members is the single most important element for building and maintaining teams. Without effective communication, teams cannot jell and cannot carry out their work.

B4.2 Elements of effective team communication

Effective team communication provides a mechanism and a venue for an open exchange of information. It is important to foster an environment in which team members can interact with each other, the team leader, the coach, and management. Among the elements that promote effective team communication are

- frequent interaction among team members, the team leader, and the coach
- regular interaction with management
- resolving misunderstandings or disagreements as quickly and amicably as possible
- maintaining a trusting team working environment in which people are encouraged to voice their opinions without fear of retribution

B5: Managing Team Project Focus

B5.1 Managing team priorities

The team leader has the responsibility for keeping the team focused on its top priorities. Therefore, the team leader should never make unilateral decisions that would affect those priorities, and should ensure that both the entire team and management understand how any requested changes will impact the work. This applies to any changes, such as design methods,

new tools, requirements enhancements, schedule acceleration, or any other element that is not included in the team's current plan.

B5.2 Managing with short term goals

The team leader should help the team members to focus on short-term goals, not just the phase, cycle, or project goals. The team leader can ensure that the team has an on-going short-term goal related to meeting upcoming deadlines and milestones in order to maintain progress towards the overall goals and schedule commitments. This helps team members to stay focused when difficulties or crises arise; by reminding the team that schedules slip one day at a time, the frequency and scope of schedule slips can be reduced, and timely actions can be taken to recover when schedule slips occur.

B6: Managing Team Roles

B6.1 The team member role

For optimal performance as a member of a TSP team, each team member should be able to answer the following questions.

- Does my individual task plan reflect the work I am currently doing and plan to do next?
- Do tasks need to be added, deleted, renamed, reassigned, re-estimated, broken down into sub-tasks, or combined?
- Does my current task plan meet commitments made to the team?
- Are all completed tasks marked accordingly?
- Do I need help from the team?
- Am I producing quality products that will meet the team's goals and expectations? If not, what remedial actions do I need to take?
- Am I following the process? If not, what remedial actions do I need to take?
- Am I recording all my time and defect data? If not, what remedial actions do I need to take?
- Have I updated my personal review checklists based on defects found?
- Have I conducted a performance analysis of my data?
- Have I set improvement goals based on past performance?
- Have I made changes to my personal process or submitted PIPs against the team processes in order to accomplish set goals?

B6.2 The planning manager role

In performing the planning manager role, the team member should be able to address questions such as the following.

- Is each team members' plans sufficiently detailed?
- Do these plans accurately represent the work that the team members are currently doing?
- If any of the team members' plans do not represent their current work, what actions do you recommend?
- If the team's workload reasonably is not well-balanced, what actions do you recommend?
- If the workload with any cooperating group or team reasonably is not well balanced, what actions do you recommend?
- Are dependencies within the team and with other related groups known, properly planned for, and tracked?

- How does the management of project data compare against the project plan?
- If the project notebook is not being maintained, including WEEK (risks, action items, individual and team status, attendees, etc.) artifacts, what remedial actions do you recommend?
- If risks are not being managed and controlled, what remedial actions do you recommend?
- If meeting minutes are not being managed and controlled, what actions do you recommend?
- If status reports are not being managed and controlled, what actions do you recommend?
- If postmortem reports are not being managed and controlled, what remedial actions do you recommend?
- Is each team member's plan accurately projecting when the team member will finish? Is the TSP tool correctly populated and maintained in order to correctly predict their finish dates?
- Are there any other planning issues that the team should be aware of?

B6.3 The process manager role

In performing the process manager role, the team member should be able to address questions such as the following.

- Do the teams have defined processes for their principal activities? If not, what processes do you recommend be defined and by whom?
- Do these processes reasonable represent the way that the work is currently being done? If not, are PIPs being submitted to correct the processes?
- When team members raise process-related issues, do you encourage them to submit PIPs and how many have they submitted?
- Are the team members following the team's defined processes?
- Is management providing the support needed to get the defined processes followed? If not, what remedial actions do you recommend?
- Are all process assets being consistently stored for future reference? If not, what remedial actions do you recommend?
- Are the team's process needs and objectives understood by the team? If not, what remedial actions do you recommend?
- Do you have a defined process for handling the team PIPs? If not, what is your plan?
- Are team members accurately recording their time, size and defect data in such a way that it can be mapped back to the process being used and the associated step or phase being performed?

B6.4 The quality manager role

In performing the quality manager role, the team member should be able to address questions such as the following.

- Is the project notebook being managed and controlled? Does it contain all work products, measures, and measurement results derived from performing the planned processes? If not, what remedial actions do you recommend?
- Are the team members properly recording their defect data?
- Do they record the defect data as they do the work or after the fact?
- Are the defect data complete and of sufficient quality to permit analysis? If not, what remedial actions do you recommend?
- Are the team members using their defect data to assess the quality of their work?

- Do the team members use their defect data to regularly update their review checklists?
- Do the team members' defect data indicate that the work is of high quality? If not, what remedial actions do you recommend?
- Are the team members holding team inspections of all work products and are these inspections being done properly?
- Are the team members conducting personal reviews of all work products and are these reviews being done properly?
- Is component and / or module quality being reviewed before integration and system test?
- Do the quality of all the components and modules meet the team's quality guidelines before integration and system test? If not, what is being done to fix the quality problems?
- Do you need further support from management or the team leader in assuring quality work?
- Are there any other quality issues that the team should be aware of?

B6.5 The support manager role

In performing the support manager role, the team member should be able to address questions such as the following.

- Are physical environment standards adequate? If not, what remedial actions do you recommend?
- Does the team have suitable tools to support its work? If not, what additional tools do you recommend?
- Does the current environment support the selected validation and verification methods? If not, what remedial actions do you recommend?
- Are all team members fluent with the available development tools?
- If any team members are not fluent with these tools, what remedial actions do you recommend?
- Does the team have adequate tool support for the configuration management process? If not, what actions do you recommend?
- Is the change control board working effectively?
- Are all changes to baselined products being managed through the configuration control system?
- Have all products that should be baselined been baselined?
- Are there any other support issues that the team should be aware of?

B6.6 The customer interface manager role

In performing the customer interface manager role, the team member should be able to address questions such as the following.

- Are we being responsive to customer requests?
- Are we properly handling customer requests?
- Is every requested change being evaluated, planned, and approved before being implemented?
- Has a criteria for evaluating and accepting good requirements been defined? Is it followed?
- Is the interface between the team members and the requirements and/or systems people working properly? If not, what should the team do to improve this interface?
- Is development being delayed by the requirements work?

- Is the quality of the requirements documentation sufficiently good to guide the development work?
- Have specifications for the creation of the requirement specifications, interface requirements, market studies, and impact analysis reports been defined? Are they followed? Are they based on customer needs? If not, what remedial actions do you recommend?
- Is bi-directional traceability being maintained? If not, what remedial actions do you recommend?
- Are the requirement specifications, interface requirements, market studies, impact analysis reports and all supporting documentation being appropriately managed and controlled? If not, what remedial actions do you recommend?
- Are the right people reviewing and approving the requirements?
- Do all team members understand the environment in which the product will be used?
- Are customer needs, expectations, constraints, and interfaces well documented and understood by the team? If not, what remedial actions do you recommend?
- Has packaging of completed products for delivery to appropriate customers been defined, negotiated and agreed to by all relevant stakeholders? If not, what remedial actions do you recommend?
- Have the procedures and criteria for verification and validation been agreed to by the customer? If not, what remedial actions do you recommend?
- Are there any other customer-related issues that the team should be aware of?
- Are there any outstanding requirements issues? What are they and what is the plan for resolving them?
- Is the requirements work on schedule?

B6.7 The design manager role

In performing the design manager role, the team member should be able to address questions such as the following.

- Are the team's design methods and notations capable of producing a quality design?
- Are operational, functional, state and logic specification templates being used to capture the design?
- Do all team members understand how to use these design methods?
- If some team members are not fluent with the design methods, what remedial action do you recommend?
- Is the team's design work of high quality?
- Are adequate design verification techniques being used to find design defects early in the process? If not, what remedial actions do you recommend?
- Has a sound architecture been produced and documented?
- Is the architecture properly controlled and maintained?
- Does the architecture consider future product evolution?
- Does the design conform to the architecture?
- Is the design properly documented and maintained?
- Are the interfaces and other design dependencies with other related teams properly identified and managed?
- Are design specifications and all relevant technical data being managed and controlled? If not, what remedial actions do you recommend?

- Are operational concepts and associated scenarios included in the design specification? If not, are they captured and how are they managed?
- Have alternative design solutions been developed and evaluated in order to ensure the best design has been selected?
- Are there any open design issues? What are they and what is the plan for resolving them?
- Are there any other design issues that the team should be aware of?
- Is the design work on schedule?

B6.8 The implementation manager role

In performing the implementation manager role, the team member should be able to address questions such as the following.

- Are all of the team members fluent in the languages being used?
- If any team members are not fluent in these languages, what remedial actions do you recommend?
- Have the proper implementation standards been developed and adopted?
- Are the implementation standards being used consistently?
- Are the team members taking advantage of shared and/or reused code where they could? If not, what improvement actions do you recommend?
- What are the implementation issues? What are they and what are the plans for resolving them?
- Does the team need any help in resolving the implementation issues?
- Are there any other implementation issues that the team should be aware of?
- Is the implementation work on schedule?

B6.9 The test manager role

In performing the test manager role, the team member should be able to address questions such as the following.

- Are test plans being produced when the process requires them?
- Are these test plans complete and thorough?
- Do the team members understand how to produce suitable test plans? If not, what remedial actions do you recommend?
- Are the system test plans being reviewed when the requirements are reviewed, the integration plans when the design is reviewed, and the unit test plans when the implementation is reviewed?
- Are sufficient test facilities planned for integration and system testing?
- Are the needed test tools available?
- Do the team members know how to use the test tools? If not, what remedial actions do you recommend?
- Are the procedures and criteria for integration of product components sufficient for ensuring a quality product in put into test? If not, what remedial actions do you recommend?
- Is the integration of product components sufficient for testing? If not, what remedial actions do you recommend?
- Are all test plans, data, and results being appropriately managed and controlled in the project notebook?

- Do the test plans, procedures, environment, and results analysis demonstrate that verification and validation criteria have been met? If not, what remedial actions do you recommend?
- Is the product quality (defect level) high enough to do system or user acceptance testing?
- Are sufficient time/resources available for testing?
- Are there any other test issues that the team should be aware of?
- Is testing on schedule?

Appendix C: TSP Coaching Guidelines

C1: The TSP Coach Role

C1.1 Requirements for effective TSP coaching

Being an effective coach requires a number of qualities, including good interpersonal skills and strong organization skills. An effective coach must also have the ability to keep teams focused on the task at hand, to focus on the process rather than the content, and to suggest alternative methods and procedures without denigrating other's ideas. A good coach should have high performance standards, coupled with the willingness to insist that the team members strive to meet those standards. Other requirements include the ability to maintain personal integrity while adapting to ever-changing situations, and the ability to function well under pressure or in a stressful environment.

C1.2 Coaching versus leading

Although the coaching and team leader roles are similar in many respects, they differ in one key respect. The team leader's job is to lead the team in building a high-quality product, whereas the TSP coach's job is to use the project work to build a superior team. The shared aspects of the team leader and team coach roles include ensuring that the team members meet their commitment for producing a quality product, helping the team to set aggressive goals and work effectively to meet them, ensuring that the team members follow their defined processes, and overseeing the timely gathering and use of data.

C1.3 TSP coaching principles

Because there is no standard coaching formula, a TSP coach must be able to adjust to the dynamic needs of a team and its members while consistently conforming to a firm set of coaching principles.

- Believe that people want to do the right thing
- Build talent by learning the team members' potentials and helping them to improve
- Set high standards for superior work, and motivate and guide team members to do better when they fall short
- Focus on improvement; show team members how much better they could have done if they had followed the process a little better and convince them to follow the process better next time
- Improve in steps and celebrate every improvement
- Focus on successful completion of the project

C1.4 TSP coaching goals

The TSP coach has three primary goals.

- Ensuring that the process is followed
- Ensuring that everyone is involved in planning, working, and contributing to every part of the process
- Enabling effective communication among team members, the team leader, the customer, and management

C1.5 TSP coaching roles

To best meet the TSP coaching goals and objectives, TSP coaches must take on many different roles, including the following.

- **Process consultant –** TSP coaches act as process experts to interpret details during the launch, checkpoints, and phase, cycle or project postmortems, and provide guidance on enacting the day-to-day work processes, the weekly meetings and reports, and the role manager responsibilities.
- **Facilitator –** During meetings, TSP coaches facilitate any discussions or difficulties, employing (as needed) their understanding of the team life cycle, group working styles, communication types, and other applicable skills.
- **Process quality assurance agent –** TSP coaches act as management's process quality assurance agent by providing management with objective and appropriate levels of visibility into, and feedback on, process and associated work products throughout the life of the project.

C1.6 TSP coach teams

When several coaches work together in an organization, they can form a coaching team to plan and track their coaching work. The coaching team provides a support structure and helps coaches to anticipate and resolve problems while learning and benefiting from the support of their peers. When coaches work as a team, they can see better ways to help and guide the development teams, build on each other's successes and share and benefit from each other's data.

C2: Guidelines for Introducing TSP into an Organization

Successful introduction of TSP into an organization can be initiated only by the senior executive in charge of that organization; however, the coach can and should guide the introduction effort through the steps needed to ensure success. The TSP introduction should always start with an executive planning seminar and planning session. The TSP coach's objectives for the executive seminar are to ensure that management understands the cultural environment required for successful TSP adoption and that they accept responsibility for creating and maintaining that environment. The following guidelines are offered to help TSP coaches during the introduction process.

C2.1 Obtain management support

Once senior management has given their support, the TSP coach should work with them to choose the departments or projects in which to launch the first TSP teams. Whenever possible, the pilot teams or projects should have managers who are enthusiastic about using TSP and who understand the implications of a TSP effort and are willing to do what will be required for both the project and TSP to be a success.

C2.2 Choose an appropriate team leader

The only solid rule for choosing team leaders for pilot TSP teams is that they must support the TSP introduction effort. However, several other factors should be considered.

- The team leader should be someone who is respected by management.
- The team leader does not have to be an engineer, but if not, the leader should ensure that the team has a strong lead engineer.
- The team leader should be someone who is respected by the team members.

C2.3 Choose appropriate team members

Although TSP coaches do not usually participate in selecting team members, they may be asked for advice on team scope or membership. Some of the factors that should be considered when choosing team members include the following.

- Are all potential team members trained in PSP and TSP? If not, do they understand the following concepts?
  - What a defined process is and how one is used
  - How to record development time data
  - How to make size and resource estimates
  - How to produce detailed development plans
  - How to update detailed development plans
  - How to track progress against a plan
- Do the team members have the same areas of expertise or specialization? If they have different specializations, do they need to be on the same team as the developers?
- Are the team members committed to implementing TSP methods and processes?

C2.4 Ensure adequate coaching resources

Because the single greatest obstacle to rapid TSP deployment in an organization is the availability of coaches, management should be made aware from the project onset of the importance of building a skilled internal coaching staff. To ensure that capable individuals are willing to consider taking a coaching position in the organization, the TSP coach should encourage management to communicate that coaching is a path for career advancement. Successful coaches should be rewarded with promotion into attractive next assignments.

C2.5 Ensure management recognition of successful teams

Coaches should urge management to establish a regular program for identifying and recognizing important individual and team achievements in a visible and significant way.

C2.6 Ensure adequate coaching support for the pilot teams

First-time TSP teams generally require full-time coaching for the first two weeks following the initial launch, and at least half time thereafter until they have completed one full project cycle and conducted a first project relaunch. Pilot organizations typically do not have an internal coaching staff so the coaching will have to be provided by external coaches. If appropriate coaches are not available locally, the coach may have to conduct coaching activities remotely, or the organization will have to incur additional expense for the coach's travel and other expenses.

C2.7 Keep management informed of team progress

During the project, it is important for the coach and team leader to make frequent reports to immediate management, and periodic reports to senior management. This keeps management

informed and also helps to retain their support. At the end of the project, the coach, team leader, and team members should prepare a comprehensive report of the team's work describing how the work compared with the plan and how it compared with prior non-TSP projects. These reports should include explicit data on team progress and performance, and will help to demonstrate the benefits of using TSP.

Under no circumstances should the coach allow management to be surprised by the team's progress. If it appears that the project will be late or have other problems, get the team to share that information with management as quickly as possible.

C3: Guidelines for Launching Teams

C3.1 Secure a commitment for management participation

Management participation in launch meetings 1 and 9 is critically important. If no senior manager can attend, get someone other than the team leader who is empowered to represent management and negotiate on behalf of management. This is important for four reasons.

- Management's presence shows the team that the project is important to the organization.
- When management doesn't bother to come, it demotivates the team.
- Forcing the team to complete the entire launch and then not have anyone review and approve the plan makes the launch effort seem pointless.
- Lack of management participation suggests that management doesn't support the TSP or care about the timeliness or quality of the team's work.

C3.2 Prepare management for meeting 1

The team coach should ensure that management personnel are able to explain to the team why they want this project done and why it is important. By explaining their goals for the project, management reinforces its importance and helps to motivate the team. Management should prepare a short presentation that answers the following questions.

- What is the team being asked to accomplish this phase or cycle?
- What are the quality, schedule and cost goals?
- What are other stated goals?
- What is a minimally successful plan? How are the goals to be prioritized and what is the flexibility?
- Why is this work important? What is the strategic or big picture goal and objective?
- What resources are available to accomplish these goals?
- Why we are using TSP for this project?
- How will the goals be measured?

C3.3 Prepare the team for launch

Make sure that the team members have had the required training, or sufficient just-in-time preparation, to successfully complete all of the launch tasks. Ensure that everyone understands that the team must make their own plan that is based on data and experience; they should not blindly accept any schedule impositions, but they should try to build an aggressive plan that meets management's requirements. The team leader's job is to support the team during the planning

process and to ensure that the members do their utmost to meet management's needs. Once the team and team leader have developed the plan, they must convince management that this plan is the best way to do the work.

C3.4 Avoid exerting undue influence

Teams view coaches as experts and often accept their views without question. Therefore, coaches should guide their teams through unfamiliar steps of the various launch meeting without unduly influencing the team's choices of process, strategy, or plan. When teams are rushed, they will likely accept any authoritative view without question, so coaches should ensure that their teams take their time to put together a thorough and thoughtful plan.

C3.5 Make sure all team members are heard

While team members may disagree on key points during the launch, they usually reach a satisfactory conclusion without much coaching. The exceptions are usually where an outspoken member tries to dominate the discussions or to sway the team towards the team member's opinion. In these cases, the following techniques will usually resolve the problem.

- Stand behind the outspoken member so he or she will have trouble talking to you (the coach).
- Go around the room and ask for each team member's views in turn. Arrange the order of discussion so that the outspoken member goes last.
- Ask the team member to hold his or her comments until the others have spoken.
- Suggest that the team use available data, not individual opinions, to help make a decision.
- Impose a time limit on all comments.
- Talk privately with the dominating team member about the disruptive behavior. Do not have the discussion in front of the whole team, as this is likely to make the other members reluctant to contribute.

C3.6 Address team member issues as needed

Team member issues are rare during the TSP launch, but when they occur, they are typically of three kinds.

- One or more members do not show up for the launch, miss some meetings, or are constantly late. This is a discipline problem that must be handled promptly by the team leader. Allowing one or two members to skip or delay launch meetings will demoralize the team and create resentful feelings that can destroy it as a working unit.
- Team members are present but do not fully participate in the launch process. They may be uninterested, unsure of themselves, or naturally quiet. The coach should regularly ask these members for comments and try to build their interest, confidence, and willingness to participate.
- Occasionally, a team member will talk too much. If the team leader or other team members seem unwilling or unable to get that person to rein in the problematic behavior, follow the guidelines in C3.5.

C3.7 Address team leader issues as needed

Team leader issues are rare during the TSP launch, but when they occur, they are typically of three kinds.

- The team leader already has a plan and doesn't see why a new one is needed. This problem can be settled during the launch only if the team members insist on making their own plan. If the problem cannot be resolved at the outset, the coach should suspend the launch and address the need for a team plan with the team leader and senior management.
- The team leader is overly assertive. As long as the team leader is willing to rationally discuss alternate views, the methods for handling outspoken team members (as outlined in C3.5) should apply.
- The team leader is too quiet and either does not participate or does not act like a leader. The coach should discuss the issue privately with the team leader and then require him or her to lead most of the remaining launch meetings with help and support from the coach.

C3.8 Enforce the "no outside observers" rule

Do not allow outsiders observe the TSP team because their presence can be distracting and disruptive.

- Observers tend to inhibit the team's discussions.
- Observers may attempt to participate in the launch when they should not.
- Observer reaction to the team's preliminary plans may unfavorable influence the outcome.

C4: Guidelines for Coaching Teams

C4.1 Start right

New teams usually have lots of problems and questions in the first few weeks after the TSP launch. Good coaching during this time is critical to the team's success. Without adequate guidance, the team can easily become discouraged and revert to their pre-TSP habits. The coach must be available to answer the team's questions, help with tool issues, and ensure that the team properly follows its process and plan.

C4.2 Prepare and use a coaching plan

The coaching plan is prepared and reviewed with the team and team leader during the launch. The plan should ensure that the TSP coach performs the following key tasks.

- For the first few weeks, be available to answer the team's questions.
- Participate in team meetings for a month or more.
- Attend the first TSP team inspection and, if there are problems, attend subsequent inspections.
- Review the planning manager's initial data consolidation and resolve any issues.
- Ask the planning manager to provide copies of the team's consolidated data and each team member's data every week. Review these data promptly and discuss any problems with the appropriate team members.

C4.3 Conduct the post-launch briefing

The team must know how to use its chosen TSP support tool in order to properly record the members' time, size, and defect data. To ensure that everyone understands how to enter and interpret data using the tool, the coach must hold a post-launch briefing at the end of the launch to familiarize new team members with the TSP tool; the briefing should also include guidelines for

conducting team activities such as the weekly meetings and team inspections. The post-launch briefing should be conducted as soon as possible after the launch. It only takes a few hours, but it can save the team several days of frustration and false starts.

C4.4 Conduct the checkpoint reviews

The TSP process calls for a checkpoint review about one to three months after the first team launch and a second checkpoint after the first relaunch. If needed, follow-up checkpoints should be conducted a month or two after each relaunch. If there are problems in any of the areas covered by the checkpoint, these problems must be addressed promptly or the team's performance will suffer. The checkpoint serves several purposes.

- The primary purpose of the checkpoint is to help the team.
- The checkpoint provides the coach a chance to assess the team's process fidelity and to see if the team and all of the members are following the process.
- The checkpoint allows the coach to ensure that the team is properly using its data to assess and report on project status and quality. The coach should make sure that the team members are following their plan to do quality work and assess their quality data to determine the degree to which they are succeeding.

When conducting the checkpoint, the coach should follow these guidelines.

- Concentrate on asking open-ended questions.
- Do not publically criticize team members who may not be properly following the process, or the team will view the checkpoint as an audit and may be uncooperative or even hostile.
- The primary objective when conducting the checkpoint is to learn what the team members are thinking and to devise ways to help them address any problems they may have.

C4.5 Promptly address project work conflicts

If conflicts arise during the various phases about how to carry out the project work, these guidelines can help the coach to guide the team toward a satisfactory resolution.

- Make sure that all parties involved in the issue are present.
- Help the members agree on a definition of the issue.
- Guide the parties through defining the criteria for a suitable solution.
- Help the team members define the possible alternative solutions.
- Have the team members evaluate the solutions against the criteria.
- If the result is not immediately obvious, guide the members in defining the steps needed to resolve the remaining issues.

C4.6 Promptly address team member problems

If one or more team members are not following the process or are otherwise disruptive to the team, the coach must work with the team leader to agree upon and implement a solution.

- The first step should be to coach or counsel the problematic team member(s) toward suitable behavior.
- If behavioral interventions are ineffective, bring the issue to management's attention. Management may attempt to counsel the member or remove them from the team. In the case where counsel is ineffective and removal is not an option, management may consider having the member work as a "contract" employee who performs project tasks but is not otherwise on the team.

- In cases where disciplinary action is the likely solution, the coach should treat this as a management problem and avoid involvement whenever possible.

C4.7 Promptly address team leader problems

If the team leader does not cooperate with the TSP coach or does not require the team members to follow the TSP process, the coach will not be able to effectively work with the team. When there are problems with the team leader, the coach should attempt to resolve the problem directly. If the direct approach is unsuccessful, the coach should seek guidance from higher-level management.

C5: Guidelines for Coaching Role Managers

C5.1 Coaching role managers

The fundamental principle of role-manager coaching is that the role responsibilities are delegated team leader responsibilities. Therefore, the only effective way to guide the role managers is to work through the team leader, with the objective of ensuring that the team members are effective in performing their role manager jobs. The coach must ensure that the team leader understands the roles in order to help the team members in effectively performing their role work. The team leader's understanding can be accessed using the following questions.

- Does the team leader have a clear vision for each role?
- Does the team leader understand the pragmatic action steps each role needs to start with?
- Does the team leader understand what questions need to be asked and when?
- Will the team leader count on the team members to do the role work?

C5.2 Helping the team leader to coach role managers

If one or more team members are not performing their roles, the coach should review the situation with the team leader and agree on an approach for handling the issue. The coach should also make sure that the team leader knows how to review the role reports during the weekly team meeting.

C5.3 Handling role manager workload problems

If a team member is not handling his or her role responsibilities, it is often because of excessive job pressure. In these cases, the coach should work with the team leader to determine the probable cause of the problem and devise an appropriate solution.

- The team member may be overloaded, and the team should consider rebalancing its workload.
- The team member is using workload as an excuse to avoid tasks that he or she considers uninteresting or unimportant. The team leader should help the team member to understand that the role tasks are essential and, if not properly and promptly handled, the team will later run into more time-consuming problems.

C5.4 Handling role manager skill issues

If the team member lacks the skills to handle his or her role responsibilities, the coach should suggest that the team leader reassign that role to a better-qualified member. Thereafter, suggest that the team consider skill needs more carefully when selecting team roles.

C5.5 Handling role manager motivation issues

If the team member is not performing the role responsibilities due to a lack of motivation or interest, the coach should work with the team leader to identify the probable root of the problem and devise an appropriate solution.

- The team member is not thinking creatively about the role. In this case, the team leader or coach should help the team member to see ways to make the role tasks more interesting.
- The team member did not understand the role during role selection, and the team leader might consider a role rearrangement.
- The team member was not interested in any of the roles and only took the assignment because it was required. This indicates that the team member is unwilling to fully participate as a team member. This situation probably indicates that the team member has other, more potentially troublesome attitudes that need to be addressed.
- The team member does not see any value in the role. This is usually occurs if the team leader does not ask any questions related to the role work, or if the role work is not appreciated by the team. The team leader should make sure to ask the role manager for a report during every team meeting, or, if necessary, should help the team member to address any resistance encountered from the team.

C6: Guidelines for Assessing Team Characteristics

To effectively coach a team, the coach must start with a clear understanding of the team's current problems and performance.

C6.1 Assessing team cohesion and motivation

Teams that are cohesive and highly motivated usually have one or more of these identifying characteristics.

- The team has a single clear, well-defined, motivating goal and an obvious sense of urgency with regard to meeting that goal.
- The team members communicate freely and openly with the team leader and among themselves.
- There are "in" jokes and a sense of "us" about the team as a group.
- People volunteer for jobs and strive to meet their commitments.
- Everyone is included, and there are no separate cliques or outsiders.
- People think about the team and how to improve it as a working unit.

Conversely, some of the symptoms that a team is demotivated or lacks cohesion include the following.

- People come in late and leave early for work, and typically are late for meetings.
- The team is fragmented into multiple small cliques and working groups.
- The team members meet privately with the team leader to complain about the team, management, or the other members.
- Team members are difficult to work with and object to following the team's process or plan.
- The members seem concerned about themselves and their jobs, and have few (if any) ideas about how to improve the team as a working unit.

C6.2 Assessing process discipline

Teams with good process discipline generally exhibit one or more of the following characteristic behaviors.

- They refer to the appropriate process scripts as they work.
- They submit their weekly data on time.
- They conduct periodic personal and team project postmortems.
- They submit and process PIPs.

Team process discipline problems may be indicated if the coach observes a combination of several of the following behaviors.

- The team members do not refer to their processes as they work.
- The team members do not submit PIPs, or there is no process for handling PIPs.
- There are problems with task time data, such as one or more team members failing to submit their weekly task time data in time for the weekly team meeting, consistently low numbers of task hours or declining task hours, task times that are substantially below the task estimates, or numerous partially-completed open tasks.
- The defect data are poor, or the data is incomplete or missing from one or more team members.

C6.3 Assessing team tracking and reporting practices

A team that is appropriately tracking and reporting on its work generally demonstrates the following behaviors.

- The team meets every week to review the status of EV and task hours, key milestones, the extent (if any) of schedule exposure, and the key project risks and mitigation plans.
- The team leader regularly reports to management on project status (and the recovery plan, if the team is late), the status of key risks, and any issues requiring management attention.

C6.4 Assessing commitment to process improvement

Teams that are committed to process improvement generally conduct one or more of the following activities.

- The team holds regular launch, phase or cycle postmortems, and project postmortems.
- The members regularly discuss process issues and propose PIPs as needed.
- The team discusses the adequacy, meaning, and proper use of their quality data.
- The team periodically holds reviews of test defect data and considers preventative actions.
- The team members review their status against goals and regularly establish new and more aggressive goals at each team launch or relaunch.

C6.5 Assessing team leadership

There are no simple indicators of effective team leadership; however, the coach can generally surmise that teams which exhibit most of the following behaviors are being effectively led.

- The team is cohesive.
- The team is reasonably disciplined in following its processes.
- The team members all strive to meet the team's goals and milestones.
- The team regularly undertakes process improvement actions.

Team leader behaviors that suggest potential leadership problems include the following.

- Defensiveness about project status
- Reluctance to freely discuss EV and task hour data with the coach or management
- Reluctance to discuss team problems with the team or with the coach
- Unwillingness to work with the coach on reporting and resolving team problems
- Blaming team members for team problems
- Complaining to the team about his or her management

C6.6 Assessing quality management

A team that is effectively managing its process and product quality has team members who properly perform the following activities.

- Gather all time, size, and defect data completely and accurately
- Conduct personal reviews of the products of every development phase for every product
- Strive to meet the yield goals for every review
- Conduct team inspections of the products of every development phase for every product
- Use historical data to determine the size and composition of the group that will perform each team inspection
- Evaluate the data from each review and inspection to decide on further quality management actions
- Use the capture-recapture method (and other means) to assess the quality of every product and product element prior to releasing them to final testing
- Reinspect, rework, or replace products or product elements that do not meet the team's established quality criteria
- Regularly hold team reviews of the product defects identified in post-development testing and usage activities, and for each defect category, establish and implement defect prevention and/or early defect detection process improvements
- Establish and implement improvement actions if personal reviews, team inspections, or testing steps do not meet the established team quality criteria
- Use the data from prior work when updating or establishing a new quality plan to set realistic and challenging personal and team quality improvement goals

C7: Guidelines for Coaching Plan Management Issues

C7.1 Coaching personal plan management

The first priority of the coach is to ensure that the team members are collecting actual data, because without real data, a plan is not manageable and has little value. Members of new TSP teams generally need guidance in the following areas when starting their first project.

- Revising initial personal plans to reflect the way they actually work
- Balancing the need for detail in their short-term and long-term plans
- Breaking near-term plans into sufficiently small tasks to regularly show EV (progress)
- Handling unanticipated tasks
- Using a TSP support tool to manage their plans
- Using their growing volume of personal task time data to improve task planning accuracy

- Using their growing volume of personal task hour data to improve task hour planning accuracy
- Identifying ways to improve their personal task time performance
- Producing PIPs to address potential process-improvement areas

C7.2 Coaching team plan management

Once the individual team members have learned how to manage their personal plans, the coach should shift priority to monitoring team member planning and to coaching the team as a unit in the following areas.

- Understanding and interpreting the data in the team's WEEK report
- Assessing team status
- Estimating project completion
- Identifying ways to improve team task time performance (such as instituting quiet time, establishing dedicated team work spaces, respecting do-not-disturb signs, understanding and minimizing the effect of task dependencies, utilizing support personnel, and minimizing distractions and interruptions)
- Recognizing potential problems with uncompleted work
- Assessing plan growth and determining the causes
- Adjusting personal and team plans for anticipated growth

C7.3 Coaching team leaders on plan management

Because the team leader can have the largest impact on the effectiveness of a team's plan management, it is important for the coach to ensure that the team leader understands and leads the team effectively and knows how to handle all of the team leader's plan-management responsibilities.

C7.4 Coaching management on plan management

The coach should work with organization management to ensure that plan-management problems are addressed at the appropriate level in the organization.

- If the team leader is not fulfilling his or her plan management responsibilities and is not responding to coaching guidance, the coach should work with more senior management to resolve the problems.
- If senior management is overly directive or is not sufficiently demanding in handling team plan management issues, the coach should provide guidance on the most effective ways to manage self-directed teams.

C7.5 Providing team plan management support

The coach should work with new teams and the team leader to help them to perform their project tracking responsibilities effectively. Once the team and its leaders have learned the basics of planning management, the coach should check back periodically to ensure that they continue to manage the plan properly.

C7.6 Resolving team plan-management coaching issues

If a TSP coach is having difficulty coaching a team leader, the coach should review the situation with the team leader's management to address possible actions, such as

- the need for team leader and/or team member training
- management counseling for the team leader
- getting a different team coach

C8: Guidelines for Coaching Data Management Issues

C8.1 Conducting timely data reviews

Promptly reviewing data is generally the most effective way of ensuring that new teams are collecting meaningful and high-quality data. Because data-gathering support is time-consuming, a dedicated TSP coach must be available during the first two weeks for each new TSP team. Once the team members have been trained, spot checks are usually sufficient to maintain team data-gathering discipline.

C8.2 Checking the quality of time data

The quality of time data may be assessed by examining the time log. Problems with the data quality may be indicated if one or more of the following are noted.

- Many of the minute or second values are 00. This may indicate that the values were entered after the fact. When many of the entries are of this character, it often means that the developer is entering approximate values after the work was completed.
- There are no interruption times. Almost everyone has periodic interruption. Lack of interrupt-time entries indicates that the data is not being recorded accurately or in real time.
- The product types, process phases, and task names are not consistent.
- Excessive time is being spent on a few tasks (or products, or phases), but others have been ignored. This suggests inaccurate data, a misunderstanding on how the work being performed maps to the tasks (or products or phases), or incorrect implementation of the work.

C8.3 Checking the quality of defect data

The quality of defect data may be assessed by examining the defect log. Problems with the data quality may be indicated if one or more of the following are noted.

- Data values are missing, such as fix times, injection or removal phases, or no type values.
- There is no description for the defects.
- The defect types are improperly categorized – they are not numbered or the defect types are incorrect.
- The defect types are inconsistent with the phases in which the defects were injected or removed. An example of inconsistency would be a coding defect injected in the design phase.
- The injected phase comes after the removed phase. For example, the data indicate that defects were removed in coding and injected in code review.
- The fix times are all identical or unreasonable for the phases and types involved.

C8.4 Checking the quality of size data

The quality of size data may be assessed by examining the product plan summary. Problems with the size quality may be indicated if one or more of the following are noted.

- There are obviously missing or incorrect data values, such as a total or added and modified size value that is inconsistent with the base, modified, added, deleted, and reused values.
- The size values are round numbers. This suggests that the values were estimated rather than measured.

C8.5 Checking the quality of task-completion data

Problems with the task-completion data may be indicated if one or more of the following are noted.

- Data values are missing, such as tasks with no completion dates.
- Completion dates are inconsistent with the project schedule.

C8.6 Checking the consistency of time and task data

By examining a full set of task or project data, it is often possible to identify consistency problems in the time and task data.

- The total time-log entries for tasks or phases are not consistent with the total values for those tasks or phases entered elsewhere.
- The time-log and task entries show that the task completed is improbable; for example, code reviews are shown as happening before the product was coded.
- The task-completion dates are inconsistent with when the tasks were performed.

C8.7 Checking the consistency of time and defect data

By examining a full set of task or project data, it is often possible to identify consistency problems in the time and defect data.

- Time-log entries are not consistent with the defect-log entries; for example, the phase for defect removal is shown occurring at the same time that another phase is being performed.
- The defect types found in several phases are illogical.
- The amount of time in a defect-removal phase is inconsistent with the number of defects found and the defect fix times. An example would be a 40-minute compile phase with few (if any) defects. Another would be a test phase in which the total defect fix times exceed the time spent in that phase.

C8.8 Checking the consistency of size and defect data

By examining a full set of task or project data, it is often possible to identify consistency problems in the size and defect data.

- The defect density values for some phases are unreasonably high or low.
- The defect profile is inconsistent.

C8.9 Making provisions for data management

Because the TSP produces a great deal of data and because these data are critical to the proper performance of TSP teams, effective data-management provisions are essential.

- A proper support system is required.
- That system must be regularly backed up and rigorously change controlled.

- All data entered into the system must be validated and corrected or otherwise noted as not validated.
- The data-management responsibility must be clearly vested, either in someone in the organization or on the team.
- Suitable provisions must be made to both ensure the privacy and protection of the data and its timely access by the team and team members.

C8.10 Ensuring data privacy

With proper coaching and management support, data privacy will not be a problem. However, provisions must be made to ensure that personal team member data are not disclosed to anyone outside the team except with the agreement of the team member involved.

C8.11 Conducting data audits

The quality of the TSP data must be regularly audited. Data audits are usually conducted by the TSP coach with the planning, process and quality role managers, with the role manager focusing on the data that pertains to their respective roles. The objectives of these audits are to ensure that the data are

- Accurate – The data values properly and accurately represent the items being measured.
- Complete – The data have no significant holes or gaps.
- Consistent – The data are self-consistent.
- Current – The data are up to date.

C9: Guidelines for Data Analyses

C9.1 The pre-launch data analysis

Prior to launch, analyze organization, team, and personal data to obtain historical data for

- size, to be used for estimating guidance and plan comparison
- quality, to be used for planning guidance and comparison
- schedule, to be used for planning guidance and comparison
- resource, to be used for planning guidance and comparison

C9.2 Analyzing data for team goal setting

When setting team goals, the team members should conduct some of the following analyses.

- Examine historical task hour data to establish realistic and challenging team goals.
- Review the success history of task hour improvement efforts and identify those efforts that are applicable for this team.
- Review the available quality data and improvement actions and identify appropriate actions for the team.
- Review unimplemented PIPs and identify PIPs that are most applicable to the next project phase or cycle, or that may contribute to the team's goals.
- Examine historical data on such team operations as weekly meeting frequency and duration, adequacy and timeliness of management reporting, conformance to configuration-management procedures, milestone performance, and setting realistic team goals.

C9.3 Analyzing data for team planning

When making cost, schedule, and quality plans, team members should analyze any available data on prior phases or projects to establish appropriate parameters for size, productivity, yield, and rate, using PROBE or other methods as appropriate.

C9.4 Analyzing data to manage the team plan

The critical plan-management analyses relate to plan growth and milestone performance. Data on these factors should be regularly monitored and interpreted to decide whether or not

- the team's plan should be revised
- team priority management must be improved
- management guidance is needed on improving the rate of external change to team requirements
- the team's planning process should be revised to ensure that future plans are suitably complete

C9.5 Analyzing data to manage quality

During team operation, the following analyses should be conducted on each software product and component.

- Verification of measurement (time, size and defects) collection
- Quality profile and PQI analysis
- Yield and review rate analysis
- Test and user defect data analysis

C9.6 Analyzing data for postmortem analysis

During the postmortem, the team should

- calculate the key quality and plan management parameters for each process or sub-process used to develop work products
- summarize the results of improvement actions and their effectiveness
- evaluate team performance in meeting stated goals
- identify and prioritize improvement opportunities
- document analysis, findings, and recommendations

C9.7 Using data analyses for improving team processes

During the postmortem (or at any other time deemed appropriate), the team role managers should

- determine whether the appropriate practices and methods were used during the project
- determine whether checklists were regularly reviewed and updated as new data became available
- determine whether and by how much the various phase and process yields could be improved through improving the team's process fidelity
- prepare PIPs for the process changes that should be made to foster improved team performance

C9.8 Using data analyses to improve team coaching

The team coach should analyze team performance and compare it to the performance of other teams to determine

- which coaching activities were most effective at the team, team member, and team leader levels
- which coaching activities were most cost effective in terms of coaching time versus team performance
- what engineering activities or practices used by one team may benefit others
- what management recommendations are appropriate regarding team coaching, membership, leadership, support, or facilities

C10: Guidelines for Coaching the Team's Quality Management

C10.1 Check the quality data

The most difficult problem in team quality management is getting the members to follow the process consistently and to report their data promptly and accurately. The coach should check periodically to ensure that the team continues to follow its process and gather its data. Reviews of quality data should start immediately after the team launch and continue until the team members are accurately and completely recording their data. The coach should conduct regular reviews of the quality data with the team and each team member to help the team to improve their process discipline.

C10.2 Plan for high product quality

The coach should guide the team in achieving high levels of product quality.

- All individuals must monitor the performance of their personal work to ensure it is within all statistical control limits.
- The team should monitor the performance of all inspections against their established control limits.
- The team must monitor the performance of all tests to ensure their completeness.
- The team must apply the same care to the design and architectural work and the final packaging and release work.
- All changes in the product must be reviewed, inspected, and tested as thoroughly as possible, because small changes are far more defect-prone than new code or large modifications.

C11: Guidelines for Coaching the Team's Schedule Tracking

There are several calculations that are commonly used to estimate project completion date, including team earned value weekly rate, earned value per task hour and estimated schedule hours, and projected end date of the last person to finish. Different approaches make different assumptions about how the data can be used to make predictions about the future. The following guidelines may be useful in interpreting earned value (EV) and task hour data for both individuals and teams.

C11.1 EV and task hours on plan

When EV and task hours both are on plan, the project is likely to be on schedule, and there is no need to revise the plan or make other changes.

C11.2 EV is on plan, task hours are low

When EV is on plan and task hours low, there are three possible situations.

- The team overestimated the work.
- The team is not spending enough time on some tasks.
- The team is not accurately recording the time spent on tasks.

C11.3 EV is on plan, task hours are high

When EV is on plan and task hours are high, the team is on schedule; however, either the team has underestimated the workload or some of the team members are spending more hours doing the work than planned. If the team is putting in more time than planned, the coach should caution them that there is a limit to how long they can work this way and still be productive.

C11.4 EV is low, task hours are on plan

When EV is low and task hours are on plan, there are two possible situations.

- The team underestimated the work and they are falling behind schedule.
- The team has numerous uncompleted tasks open at the same time, and EV cannot be credited to those tasks until they are complete. This situation makes it difficult for the team to assess their actual standing against the planned schedule.

C11.5 EV and task hours are both low

When EV and task hours are both low by about the same amount, the real problem is low task hours.

- If task hours are low because team members are working only part time on the project or the team is understaffed, the team leader can use these data to support a request to management for more resources or schedule relief.
- If the team is fully staffed but unable to meet its task hour plan, the plan should be reassessed and reviewed with management.

C11.6 EV is low and task hours are high

When EV is low and task hours are high, the team has either seriously underestimated the work or had a major increase in project scope. This situation requires the plan to be reassessed as soon as possible, and a comprehensive review meeting with management.

C11.7 EV is high and task hours are low

When EV is high and task hours are low, the team is probably on schedule; however, either the team has overestimated the workload or some of the team members are not putting in as many hours as planned for doing the work. If the latter scenario is true, the team should make sure that the quality plan is being followed.

C11.8 EV and task hours are both high

When EV and task hours are both high, the team appears to be ahead of schedule and may be able to handle more work. However, the team should make sure that the quality of the work is not being compromised by the accelerated schedule.

C12: Guidelines for Coaching the Postmortem

C12.1 The postmortem objective

The postmortem is the time when TSP teams learn from their own experiences. The coach must motivate the team to use the key process data to produce a postmortem report. The team should analyze the data and use the findings as a basis for discussing how they can improve their personal and team process fidelity and the process itself in order to make further improvement. Throughout this process, the coach should dwell on the team's achievements to date and the challenges remaining, rather than criticizing poor performance.

C12.2 Using the postmortem to identify data gaps

When coaching the postmortem, the coach should focus on getting the team members to think about the data they would like to have on hand the next time they launch or relaunch a team project. The coach should help teams to gather as much data as they can from the records of the work they have just completed. The team should assess the completeness of their data, objectively reviewing what they can learn from the available data, and identifying gaps that make the data less useful. The coach should discuss these data gaps and ask the team to identify what data they would like to have at the same point in their next phase, cycle, or project. The coach should suggest that the team formulate data gathering goals for the next phase, cycle, or project.

C12.3 Using the postmortem to identify process improvement opportunities

For launch or relaunch postmortem meetings, the coach should ask for suggestions about the launch preparations, and then ask the team to identify what went well and what could be improved in each of the launch meetings. Finally, the coach should ask the team to discuss overall issues or concerns about the launch or relaunch process and how it was handled.

Phase, cycle and project postmortems should begin with a discussion of the postmortem preparation, with team members providing suggestions for improvement, as needed. The coach should then ask the team to address issues and problems with various process phases and activities. Also review the role manager responsibilities, process fidelity, the use of data, and the overall team performance against goals. Close the postmortem discussion by asking if anyone had any overall issues or concerns about the postmortem process and how it was handled.

C12.4 Using the postmortem to identify individual improvement opportunities

Ask the team members to think of any ways or situations in which coaching support might have been more effective. The coach should also suggest that the team leader and team members ask for comments or suggestions about how they handled their parts of the process.

C13: Guidelines for Coaching TSP Multi-teams (TSPm)

C13.1 TSPm launch preparation

The principal steps in preparing the leadership team for a TSPm launch are as follows.

1. Purpose – Define the leadership, structure, and responsibilities of the teams in sufficient detail so that the launch can be run as parallel TSP team launches.
2. Resources – Allocate resources to the teams.
3. Training – Train all of the team members.

4. Coaching – Obtain a lead coach and the coaches for each team.
5. Conceptual design – Form a working group of the lead designers from each team and have it produce the conceptual design.
6. Strategy – The leadership team produces a preliminary project strategy.
7. Product assignments – The leadership team assigns the elements of the conceptual design to the teams.
8. Assign role team mentors – The leadership team members each act as mentors for one or more of the TSPm role manager teams.
9. Strategy presentation – The leadership team prepares a strategy presentation to the entire team for launch meeting 1A. This presentation explains the goals, strategy, preliminary team product allocations, and team resources.

C13.2 Cautions for dealing with the leadership team

Leadership teams may try to do too much during launch preparation, such as preparing an overall project plan. If the leadership team tries to define the products each team is to develop, the resources for doing the work, and the dates when the work is to be completed, the leadership team has, in effect, produced the project plan. This undermines the individual teams and they will not work as self-directed teams. Therefore, the coach must make the leadership team understand that, although some level of planning is required before the leadership team can assign resources to the teams, the preparation work must be limited to defining what the teams must do and the resources allocated for that work. The actual task definition, size estimation, and schedule planning must be left to the teams.

C13.3 Mentoring TSPm role manager teams

The role managers are usually confused about what to do in their weekly role manager team meetings. The recommended approach is to have one member of the leadership team act as a mentor for each role manager team. The mentor will explain the tasks that the leadership team has delegated to that role manager team and provide guidance on how to handle the tasks. The role manager teams should report weekly to the leadership team on their progress.

C13.4 Coaching the TSPm launch

Coaching teams in a multi-team launch is much like coaching a single TSP team. The lead coach is responsible for coaching the leadership team and coordinating the launch activities of the teams with the team coaches. The lead coach schedules the role manager team meetings and schedules and facilitates the leadership team meetings. The lead coach may assist team coaches if needed.

C13.5 Coaching TSPm team operation

The standard coaching guidelines apply to coaching a TSPm multi-team, with emphasis on

- cross-team communication
- role manager team mentoring
- role manager team meetings
- workload balance
- crowd control
- unit and overall team tracking and reporting

C13.6 Coaching distributed multi-teams

As with the standard TSPm, every team must have the support of a readily available coach. In addition, the leadership team also must have a qualified multi-team coach readily available to support and guide its work. In addition to the normal multi-team issues, the lead coach must give special consideration to building team relationship, launching teams different geographic locations, guiding the distributed role manager teams, and building the leadership team.

C14: Guidelines for Coaching Other TSP Team Types

C14.1 Coaching functional teams

When coaching functional teams, the coach should be aware of areas that may need special attention.

- **Team integration** – The principal challenge is to get the team to feel like one coherent and interdependent group, and then to retain that feeling of oneness.
- **Team member coaching** – Since the members generally will be working alone or in very small sub-teams, coaching the individual team members is more important for functional teams than for project teams.
- **Tracking** – Although the standard TSP tracking methods apply for functional teams, multiple short-term milestones and frequent team leader status reviews also may be needed to maintain team focus and productivity.
- **Status reporting** – Status reporting is more important for functional teams than for project teams. Since functional teams typically provide support capabilities, management rarely understands their importance or can see any evidence of their effectiveness. Regular and factual reports can help to address this problem.
- **Task priorities** – Since the members of functional teams typically have many assigned tasks, task priority and workload balancing are especially important.

C14.2 Launching integrated teams

When launching an integrated team, the goals discussion in meeting 2 is particularly important. When guiding this discussion, the coach and team leader should consider the following guidelines.

- Be alert for communications problems; the same terms may mean different things to different specialties.
- If groups have trouble agreeing on overall goals, consider broadening the problem and considering the overall business, the customer's needs, or even the national objective.
- When agreement has been reached on overall goals, refine these goals to be more specific by stakeholder and specialty group. If specialty groups cannot agree on a common set of key team goals, split the teams into separate sub-teams.
- Once the goals have been established and agreed to by all team members, the rest of the launch can proceed according to the TSP launch scripts.

C14.3 Coaching integrated teams

Even with different processes and methods, integrated teams usually work well together if they can establish common terminology and a shared understanding and commitment to the project. To facilitate efficient teamwork, the coach should ensure that all team members have a shared understanding of the work to be done, use a common set of processes and methods and common terminology, and have the ability to work in an interdependent team environment.

C14.4 Launching distributed teams

Wherever possible, bring the distributed team together in one location for the team launch. The launch process then follows the standard process for launching a TSP team.

Appendix D: TSP Team Leader Guidelines

D1: The Team Leader Role

D1.1 The team leader's role during the launch

The behavior and effectiveness of the team leader is the single most important factor in determining that team's performance during a TSP launch. To be an effective leader, the team leader must work closely with the coach to

- ensure that the team has clear and compelling goals for every launch activity
- maintain the team's morale, energy, drive, and sense of urgency throughout the launch
- require all team members to follow the defined processes for each step of the launch
- ensure that every team member's views are heard and considered, that all members' opinions are honored, and that everyone's individuality is respected
- ensure that the team's decision process always produces a proper and a consensus conclusion
- resolve team issues and make any needed decisions
- establish and maintain an attitude of mutual trust and commitment
- encourage timely, open, and full communication among all team members
- recognize and complement the team for its accomplishments during the launch

D1.2 The team leader's role in delegating team tasks

When delegating tasks to team members, the team leader should allow them do the tasks but periodically check to see if the individuals need help.

- During the team launch, the team members select their own roles. The team leader may influence the role-selection process, but he or she must not assign the roles to team members.
- The team leader should make sure that the team members know that their job is to think broadly about their role responsibilities and to act as if they were running the project in all aspects that relate to that role. The role manager responsibilities include acting as the team's conscience in the particular area involved.
- The team leader should think about tasks that must be done and ask the role managers to handle them.
- The team leader should check periodically to make sure that the assigned role tasks are being done.

D1.3 The team leader's role as an engineer

While it is generally best for the team leader to devote all of his or her energy and time to leading and guiding the team, it is sometimes appropriate for the team leader to handle development work, particularly on small teams. However, when making such decisions, it is important to remember that the team leader's job is to guide and lead the team, not to do the work. When the team leader performs some of the team member tasks, the leader is likely to become enmeshed in the details of doing the work, rather than leading the work.

D1.4 The team leader's role as a positive role model

The team leader should model the behaviors that he or she would like to see in each team member. If the team leader does not act as a positive role model, his or her ability to build, motivate, and maintain high-performing teams is undermined. Examples of counter-productive actions that the team leader should try to avoid include the following.

- Showing bias toward one tool or technique
- Creating the impressing that one person is favored over other team members
- Doing the work for the team instead of holding them accountable for doing it
- Allowing distrust or disrespect to occur between team members
- Failing to creating a sense of purpose and urgency for the team
- Speaking in non-neutral terms or stating an opinion before understanding the data or the team member's perspective
- Making commitments on behalf of the team before consulting the team

D2: Team Leader Guidelines for Plan Management

D2.1 The team leader's plan-management responsibility

With TSP, it is the team who develops, commits to, and manages their plans. Nevertheless, the team leader has a key role in plan management.

- As part of the team, the team leader participates in producing the plan and personally commits to accomplishing the work in that plan.
- As team leader, he or she is responsible to management for ensuring that the plan is sound and that it will be properly executed.

D2.2 The team leader's role in making a team plan

When leading a team in making a plan, the team leader must maintain a careful balance between motivation and achievability. To be motivating, a team plan must be challenging and require that the team work aggressively to meet its commitments. To be achievable, the plan must cover all of the known work, allow at least as much time for the work as the team members have historically taken for similar work with a modest allowance for unknown additional activities, and based on a conservative estimate of the available task time for each team member.

D2.3 Managing team priorities

The team leader's job is to keep the team focused on its priorities. The team leader should meet regularly with the team members to discuss what they are doing and to keep them focused on their most important immediate tasks. The team leader should also review task status at every weekly team meeting and identify any dependency or workload imbalance problems. When the team achieves key milestones, the team leader should recognize and reward the achievement.

D2.4 Helping the team to meet their commitments

To help keep the team members focused on meeting their commitments, he or she should regularly and precisely track team progress against the plan. If the schedule has slipped, even by one day, ask the team to develop a recovery plan, and review their progress until the team is back on track. If it appears that recovery actions will require management help, discuss the problems with management at the earliest possible time and identify specific ways for management to help.

D3: Team Leader Guidelines for Quality Management

D3.1 Managing quality

The team leader delegates to the quality manager the responsibility of monitoring the team's work and of alerting the team and team leader to quality problems. The team leader should ensure that the quality manager makes a quality assessment of every product before it is released. If a product appears to have defects, the quality manager should alert the team and get the problems fixed. The team leader should monitor the quality manager's performance and ensure that he or she consistently performs the assigned responsibilities.

D3.2 Managing quality assurance

In dealing with the quality assurance (QA) group, the team leader should explain that the team's quality manager will serve as the team's interface to the QA group, and that the team counts on the QA group for quality advice and assistance. QA people can help TSP teams by analyzing team data regularly and alerting the quality manager to potential problems. QA staff must understand and respect individual data privacy when reporting issues to management.

D4: Developing the Team

D4.1 Maintaining team communication

When team members communicate openly and completely, they develop an intimate understanding of what the other members think, believe, and feel. There is no single strategy for ensuring open and full team communication, but a principal requirement is close and continuous interaction among all team members. The team leader should encourage team members to interact with one another, and, if necessary, create opportunities for them to do so. The weekly team meeting is an important part of maintaining team communication

D4.2 Holding team meetings

A properly run team meeting provides the sense of membership and the feedback needed to maintain team commitment. As part of a properly run team meeting, the team leader should emphasize the importance of following the team process and check that the members are properly gathering their process data. He or she should also ensure that all team roles are being properly performed and ask for regular reports.

D4.3 Resolving team problems

To maintain team cohesion, the team leader must require the team members to work together to solve their problems. He or she should also

- refuse to handle team issues without involving the entire team
- give the team the time and opportunity to work out its own problems
- meet with all of the involved team members to resolve issues
- support team decisions

D4.4 Resolving motivation problems

If the team leader becomes aware that the team is having motivation problems, he or she should meet with the team to discuss the issue. The team leader should start the meeting by admitting that he or she accepts the responsibility for any team working problems. Then, the team leader should

- focus the discussion on actions that were or were not taken and keep the discussion about process, and not personality or style
- review possible improvement actions and ask for the team members' comments and views
- ask the team members to think about the issues and make additional suggestions at any time

If personnel-related actions are required, the team leader *must* ensure that these actions are documented and agreed to by his or her immediate manager *before* acting on or discussing them with the team or any team member.

D4.5 Developing team members

To maximize team performance, the team leader must ensure that all of the team members are assigned to tasks that suit their personal interests and capabilities, and provide them with opportunities to hone their skills. He or she should also recognize that most people are capable of doing far more challenging work than they are currently doing, and encourage the team members to develop new skills whenever possible.

D5: Protecting the Team

D5.1 The team leader's dual responsibilities

The team leader is part of the management team and also part of the technical team. He or she is responsible to management for the team's performance and also responsible to the team for management's actions. If management tries to ask the team to perform tasks that do not relate to the team project, and that request is likely to jeopardize the team's ability to do its job, the team leader should tell management about that impact before agreeing to do the requested task.

D5.2 Balancing priorities

The team leader's principal job is running the project and using the team to do the work. Even though the team is the leader's full-time assignment, management will occasionally ask that leader to take on other special assignments. The team leader should treat every special assignment request in exactly the same way as a requirements change: assess the impact of the requested work, make that impact known before agreeing to do the work, and attempt to help whenever it will not jeopardize the project.

D5.3 Maintaining data confidentiality

The team leader must protect the confidentiality of team member data. He or she must never provide data on any individual team member to anyone. The team leader should also advise the team members not to provide personal data to anyone other than the team members or the team coach.

D6: Working with the TSP Coach

Strong and effective leadership is the single most important requirement for team success, but effective coaching is a strong second. Therefore, the team leader should

- work closely with the coach and monitor the coach's performance against the coaching plan
- ask regularly for the coach's views on how the team is performing, and where and how it can improve
- be open to the coach's feedback on how he or she can improve as a leader to better meet the needs of the team and its team members
- consider using the coach to facilitate any meeting in which the team has been asked to make an important commitment, or decision, in order to ensure that the team follows a sound decision process and not unduly influenced by any one person
- ensure that the coach is performing checkpoints, phase, cycle, or project postmortems, and working with individual team members as they strive to improve their individual performances on a regular basis.

Acronyms Used

BOK	Body of knowledge
COQ	Cost of quality
DRL	Defect-removal leverage
EV	Earned value
IEEE	Institute of Electrical and Electronics Engineers
PIP	Process improvement proposal
PSP	Personal Software Process
PROBE	Proxy-based estimation
PQI	Process quality index
PV	Planned value
TSP	Team Software Process
TSPd	TSP distributed team
TSPf	TSP functional team
TSPI	Integrated TSP team (also called *TSP integrated team*)
TSPi	introductory TSP team (also called *TSP academic team*)
TSPm	TSP multi-team
TSP+	TSP extension used by teams in organizations that are also implementing CMMI
SEI	Software Engineering Institute

References

URLs are valid as of the publication date of this document.

[Berne 1966] Berne, Eric. *The Structure and Dynamics of Organizations and Groups.* New York: Grove Press, 1966.

[Burns 1978] Burns, James MacGregor. *Leadership*. New York: Harper & Row, Publishers, Inc., 1978.

[Carleton 2010] Carleton, Anita; Over, Jim; Schwalb, Jeff; Kellogg, Delwyn; Chick, Timothy A. *Extending Team Software Process (TSP) to Systems Engineering: A NAVAIR Experience Report (CMU/SEI-2010-TR-008).* Pittsburgh, PA: Software Engineering Institute.

[Chick 2006] Chick, Timothy A. "Using TSP With a Multi-Disciplined Project Management System," *CrossTalk, 19,* 3 (March 2006): 4-8.

[Chick 2009] Chick, Timothy A; Cannon, Robert; McHale, James; Nichols, William; Pomeroy-Huff, Marsha; Welch, Jefferson; Willett, Alan. *Team Software Process (TSP) Coach Mentoring Program Guidebook (CMU/SEI-2009-SR-009).* Pittsburgh, PA: Software Engineering Institute.

[Conner 1982] Conner, Darrell R. & Patterson, Robert W. "Building Commitment to Organizational Change," *Training and Development Journal*, *36,* 4 (April 1982): 18-30.

[Constantine 1993] Constantine, Larry L. "Work Organization: Paradigms for Project Management and Organization." *Communications of the ACM 36*, 10 (October 1993): 34-43.

[Davenport 2005a] Davenport, Thomas. "Knowledge Workers Need Better Management." *CIO Insight*, 5 August 2005. http://www.cioinsight.com/c/a/Expert-Voices/Knowledge-Workers-Need-Better-Management/

[Davenport 2005b] Davenport, Thomas. *Thinking for a Living: How to Get Better Performance and Results from Knowledge Workers*. Boston, MA: Harvard Business School Press, 2005.

[DeMarco 1987] DeMarco, Tom & Lister, Timothy. *Peopleware: Productive Projects and Teams.* New York: Dorset House Publishing, 1987.

[Douglass 2009] Douglass, Bruce Powel. *Real-Time Agility: The Harmony Method for Real-Time and Embedded Systems Development.* Upper Saddle River, NJ: Addison-Wesley Professional, 2009.

[Drucker 1999] Drucker, Peter F. *Management Challenges of the 21st Century*. New York: Harper Business, 1999.

[Humphrey 1995] Humphrey, Watts S. *A Discipline for Software Engineering*. Reading, MA: Addison-Wesley, 1995.

[Humphrey 1997] Humphrey, Watts S. *Introduction to the Personal Process*. Reading, MA: Addison-Wesley, 1997.

[Humphrey 2000] Humphrey, Watts S. *Introduction to the Team Software Process*. Reading, MA: Addison-Wesley, 2000.

[Humphrey 2005] Humphrey, Watts S. *PSP: A Self-Improvement Process for Software Engineers*. Reading, MA: Addison-Wesley, 2005.

[Humphrey 2006a] Humphrey, Watts S. *Coaching Development Teams*. Reading, MA: Addison-Wesley, 2006.

[Humphrey 2006b] Humphrey, Watts S. *Leading a Development Team*. Reading, MA: Addison-Wesley, 2006.

[Humphrey 2006c] Humphrey, Watts S. *TSP Executive Seminar.* Pittsburgh, PA: SEI, 2006.

[IEEE 2004] IEEE Computer Society. *Guide to the Software Engineering Body of Knowledge (SWEBOK) 2004 Version.* http://www.swebok.org/home.html (2004).

[Kidd 1994] Kidd, Alison. "The Marks are on the Knowledge Worker." In *Proceedings Companion of CHI, 1994* (Boston, MA, April 24-28, 1994). ACM, New York, 1994, pp. 186-191.

[McAndrews 2000] McAndrews, Donald. *The Team Software Process (TSP): An Overview and Preliminary Results of Using Disciplined Practices (CMU/SEI-2000-TR-015)*. Pittsburgh, PA: Software Engineering Institute.

[Pomeroy-Huff 2009] Pomeroy-Huff, Marsha; Cannon, Robert; Chick, Timothy A.; Mullaney, Julia; & Nichols, William. *The Personal Software Process Body of Knowledge (PSP BOK)*, *Version 2.0 (CMU/SEI-2009-SR-018).* Pittsburgh, PA: Software Engineering Institute.

[Rogers 2003] Rogers, Everett M. *Diffusion of Innovations (Fifth Edition)*. New York: Free Press, 2003.

[Tuckman 1965] Tuckman, Bruce. "Developmental Sequence in Small Groups." *Psychological Bulletin 63*, 6 (June 1965): 384-399.

[Williams 1995] Williams, Ron. "Self-Directed Work Teams: A Competitive Advantage." *Quality Digest 15*, 11 (November 1995).

[Zaccaro 1991] Zaccaro, S. (1991). "Nonequivalent Associations Between Forms of Cohesiveness and Group-related Outcomes: Evidence for Multi-dimensionality." *Journal of Social Psychology 131*: 387-399.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. **AGENCY USE ONLY** (Leave Blank)	2. **REPORT DATE** July 2010	3. **REPORT TYPE AND DATES COVERED** Final
4. **TITLE AND SUBTITLE** Team Software ProcessSM (TSPSM) Body of Knowledge (BOK)		5. **FUNDING NUMBERS** FA8721-05-C-0003
6. **AUTHOR(S)** Watts S. Humphrey Timothy A. Chick William Nichols Marsha Pomeroy-Huff		
7. **PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)** Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213		8. **PERFORMING ORGANIZATION REPORT NUMBER** CMU/SEI-2010-TR-020
9. **SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)** HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116		10. **SPONSORING/MONITORING AGENCY REPORT NUMBER** ESC-TR-2010-020
11. **SUPPLEMENTARY NOTES**		
12A **DISTRIBUTION/AVAILABILITY STATEMENT** Unclassified/Unlimited, DTIC, NTIS		12B **DISTRIBUTION CODE**

13. **ABSTRACT (MAXIMUM 200 WORDS)**

The Team Software Process Body of Knowledge (TSP BOK) was drafted to define the fundamental knowledge and skills that set TSP-trained individuals apart from other software professionals. It helps individual practitioners to assess and improve their own skills, provides employers with an objective baseline for assessing the process improvement skills and capabilities of their development team members, and guides academic institutions that want to incorporate TSP into their software and other engineering courses or curricula. The TSP BOK also facilitates the development of TSP certification programs that are based on a well-established standard set of knowledge and skills.

14. **SUBJECT TERMS** Team Software Process, Body of Knowledge, PSP, TSP, BOK, CMMI	15. **NUMBER OF PAGES** 148
16. **PRICE CODE**	

17. **SECURITY CLASSIFICATION OF REPORT**	18. **SECURITY CLASSIFICATION OF THIS PAGE**	19. **SECURITY CLASSIFICATION OF ABSTRACT**	20. **LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UL