# Trabajo disciplinado en ingeniería de software (TSP, PSP)

Diego Vallespir

# Aclaración

La mayoría de las slides provienen del curso del Personal Software Process del SEI-CMU

# Why Projects Fail?

# Why Projects Fail?

Project commitments are often unrealistic.

Large projects are harder to control.

Quality problems.

Lack of good leadership and coaching.

And more…

# What is the PSP?

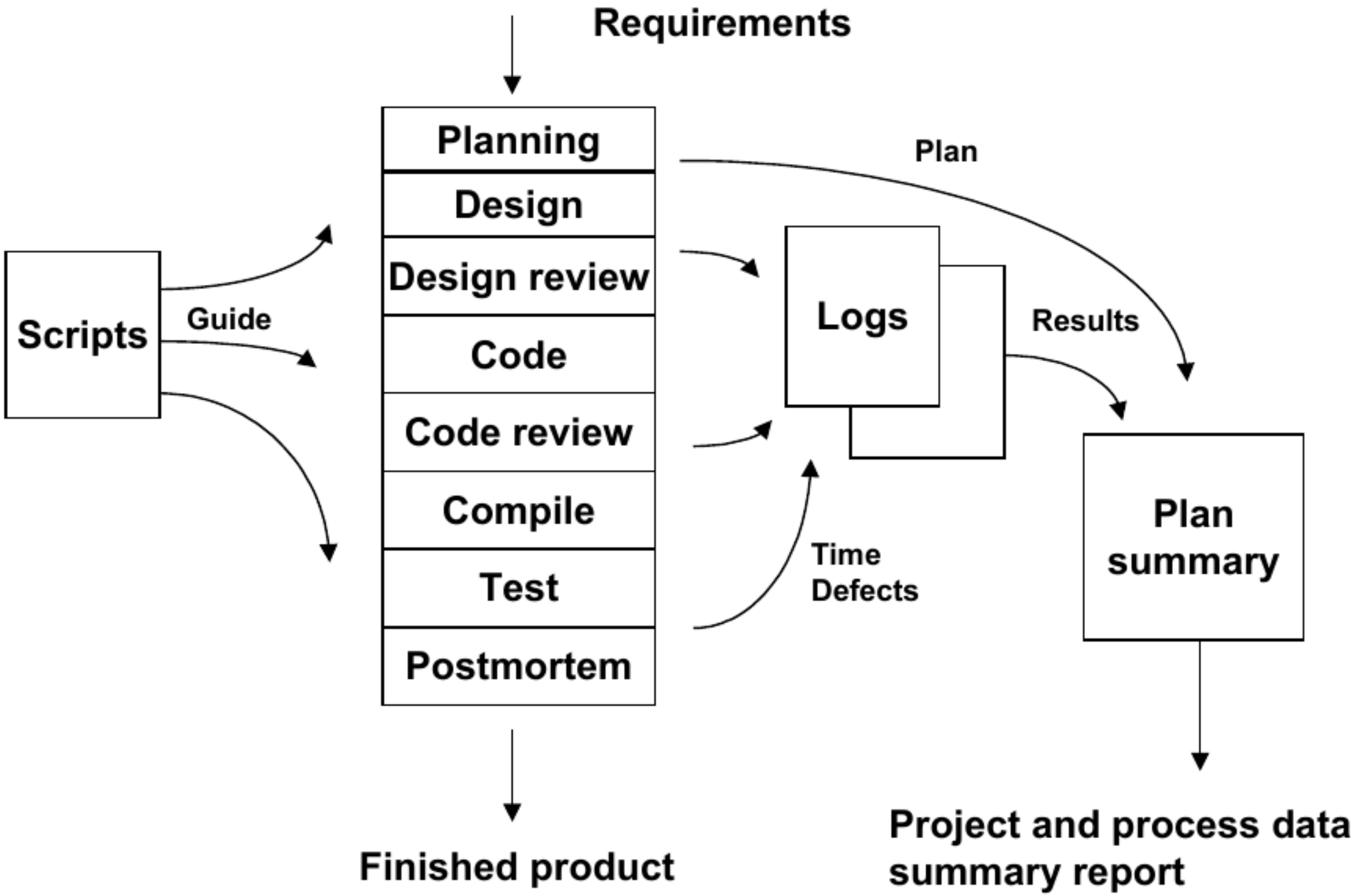
The PSP is a personal process for developing software or for doing any other defined activity.  The PSP includes

- defined steps
- forms
- standards

It provides a measurement and analysis framework for characterizing and managing your personal work.

It is also a defined procedure that helps you to improve your personal performance.

# The PSP Process Flow



# The PSP Measurements

The basic PSP data are

- program size
- time spent by phase
- defects found and injected by phase

Both actual and estimated data are gathered on every item.

Measures derived from these data

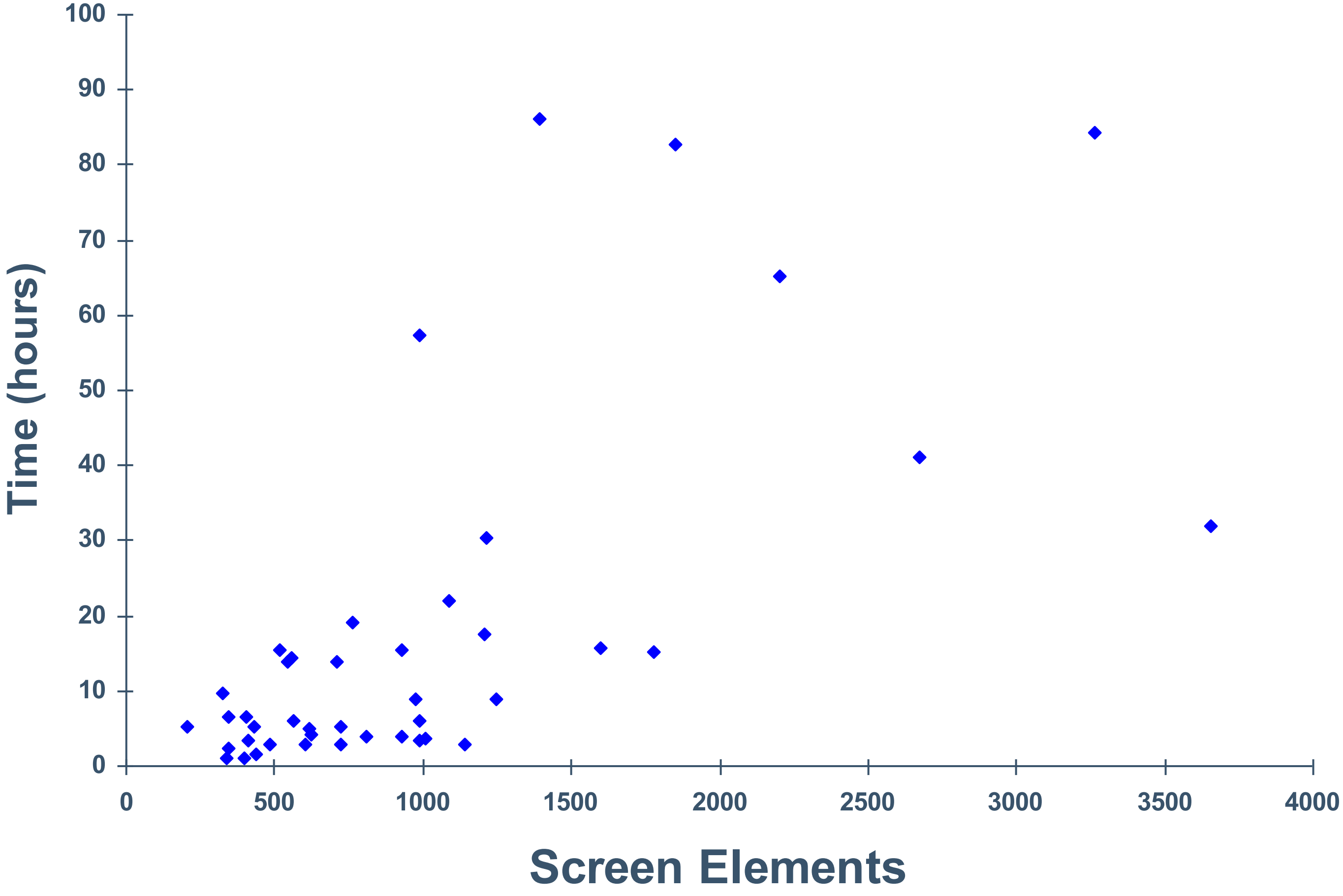
- support planning
- characterize process quality

# Size Measurement Criteria

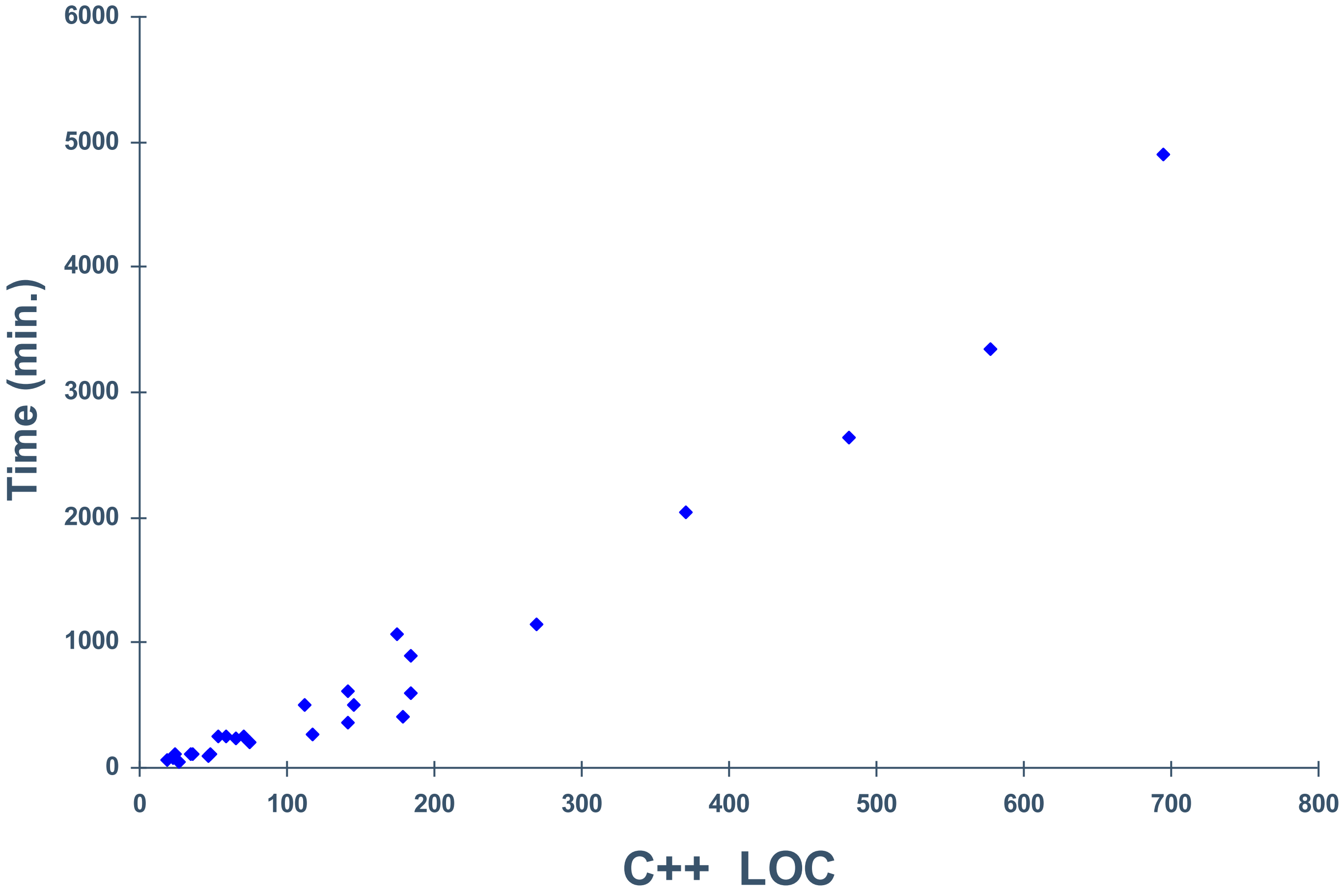
Size measurements must be

- related to development effort
- precise
- machine countable
- suitable for early planning

# Screen Elements Versus Time



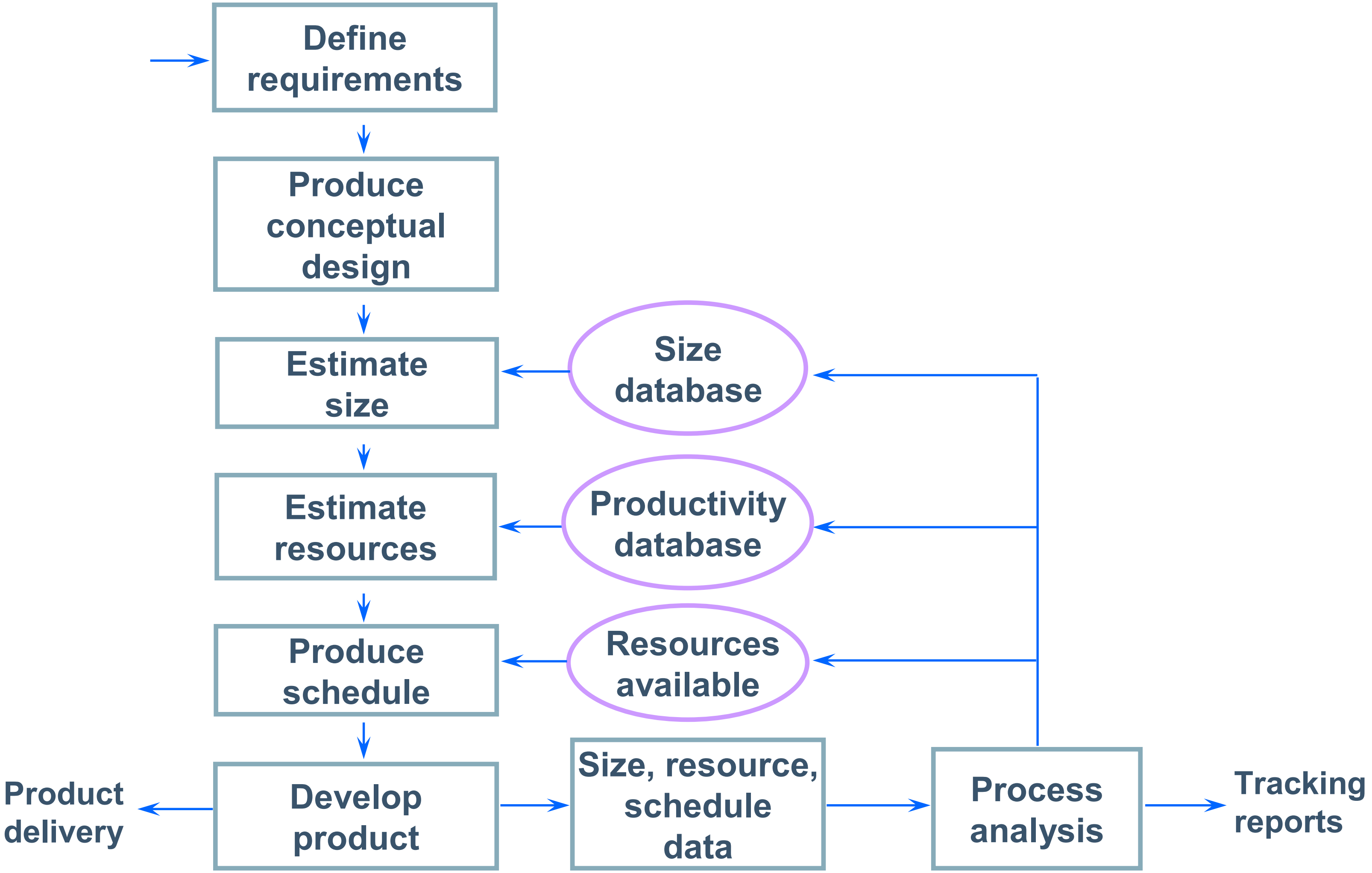
# C++ LOC Versus Time



# Planning Overview

- Plans
  - allow you to make commitments that you can meet
  - provide the basis for agreeing on job scope, schedule, and resources
  - guide the work
  - facilitate progress tracking and reporting
  - help ensure that key tasks are not overlooked

# The Project Planning Framework



Define requirements

Produce conceptual design

Estimate size

Size database

Estimate resources

Productivity database

Produce schedule

Resources available

Develop product

Product delivery

Size, resource, schedule data

Process analysis

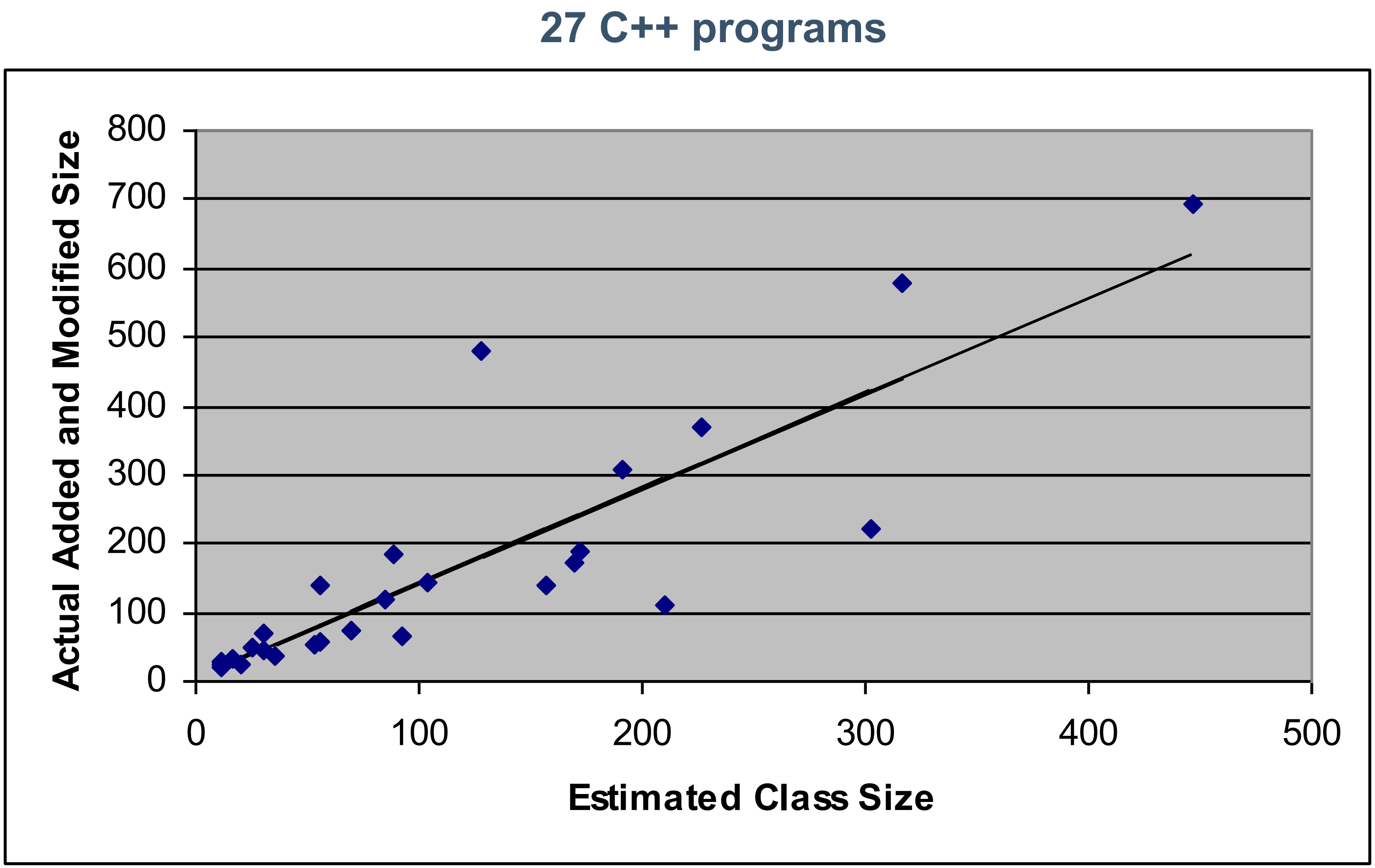
Tracking reports

# Size Estimating Principles

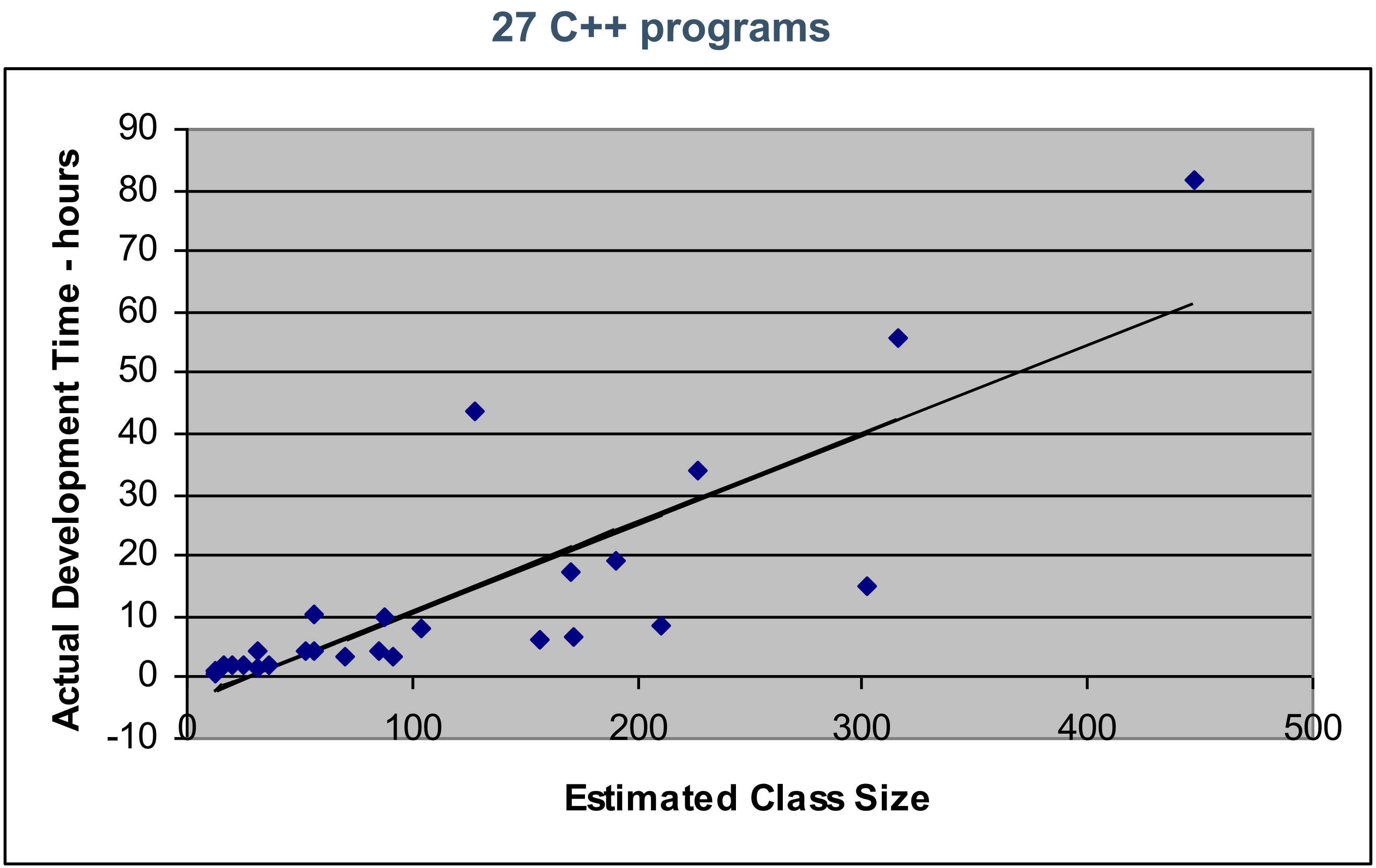
- Estimating is an uncertain process.
  - No one knows how big the product will be.
  - The earlier the estimate, the less is known.
  - Estimates can be biased by business and other pressures.

- Estimating is an intuitive learning process.
  - Ability improves with experience and data.
  - Some people will be better at estimating than others.

# Regression Line for Program Size



# Regression Line for Development Time



# The PSP Quality Focus

The defect content of software products must be managed before more important quality issues can be addressed.

Low defect content is an essential prerequisite to a quality software process.

Since low defect content can best be achieved where the defects are injected, engineers should
remove their own defects
determine the causes of their defects
learn to prevent those defects

# The Economics of Quality

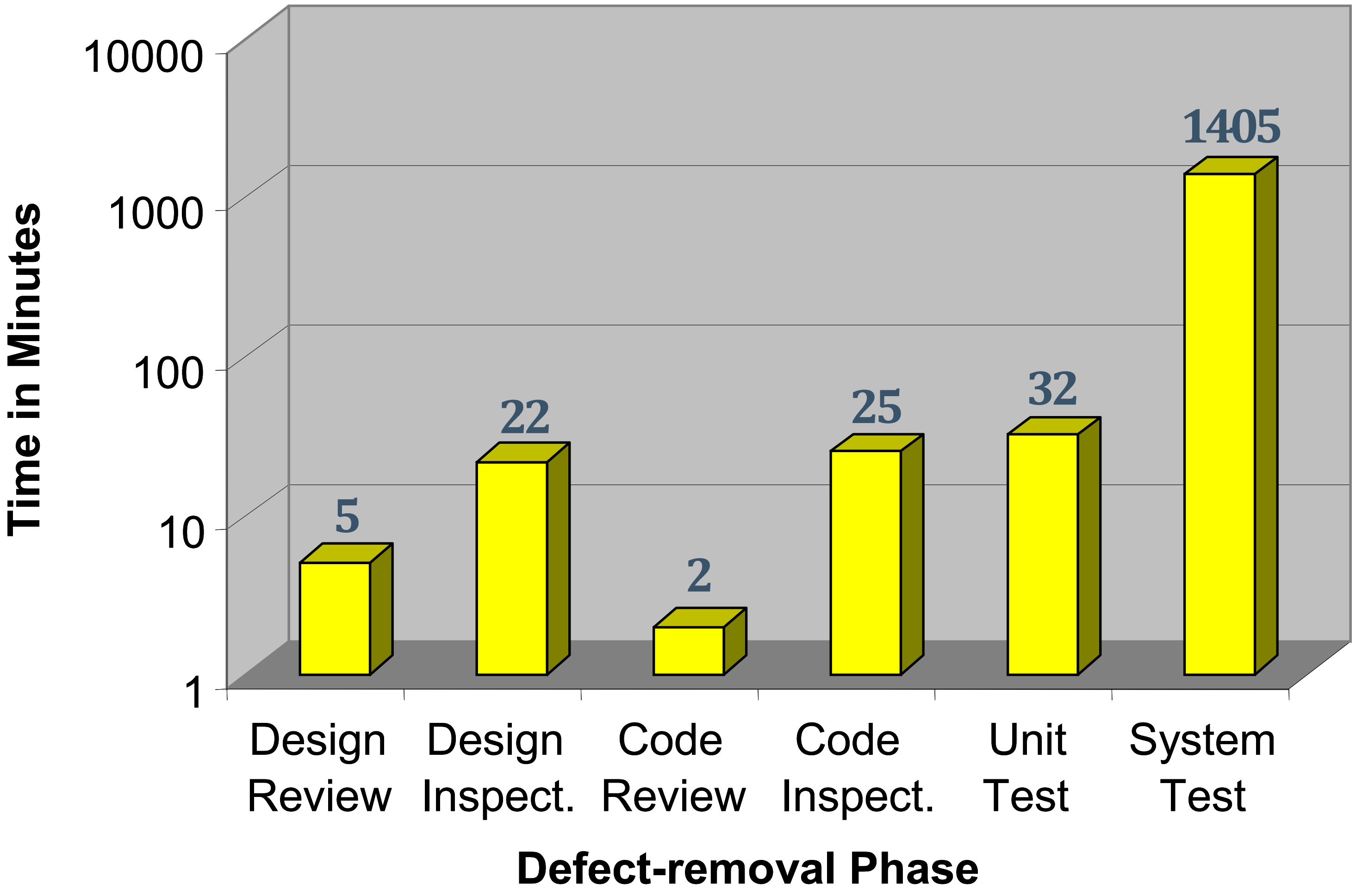
Software is the only modern technology that relies on testing to manage quality.

With common quality practices, software groups typically
spend 50+% of the schedule in test
devote more than half their resources to fixing defects
cannot predict when they will finish
deliver poor-quality and over-cost products

To manage cost and schedule, you must manage quality.

To get a quality product out of test, you must put a quality product into test.

# Defect-removal Times



Source: Xerox

# Defect-removal Rates

Even at the personal level, it is more efficient to find defects in reviews than in testing.

Unit test finds only about 2 to 4 defects per hour.

Unit test finds about 50% of the defects.

Code reviews find about 6 to 10 defects per hour.

Practiced reviewers can find 70% or more of the defects before compiling or testing.

# Design Review Principles

In addition to reviewing code, your should also review your designs.

This requires that you

- produce designs that can be reviewed
- follow an explicit review strategy
- review the design in stages
- verify that the logic correctly implements the requirements

# Quality Measures

To do efficient reviews, you must have measures.

The PSP has many useful quality and process-control measures.

yield

review rate

defects found per unit of size

defects injected and removed per hour

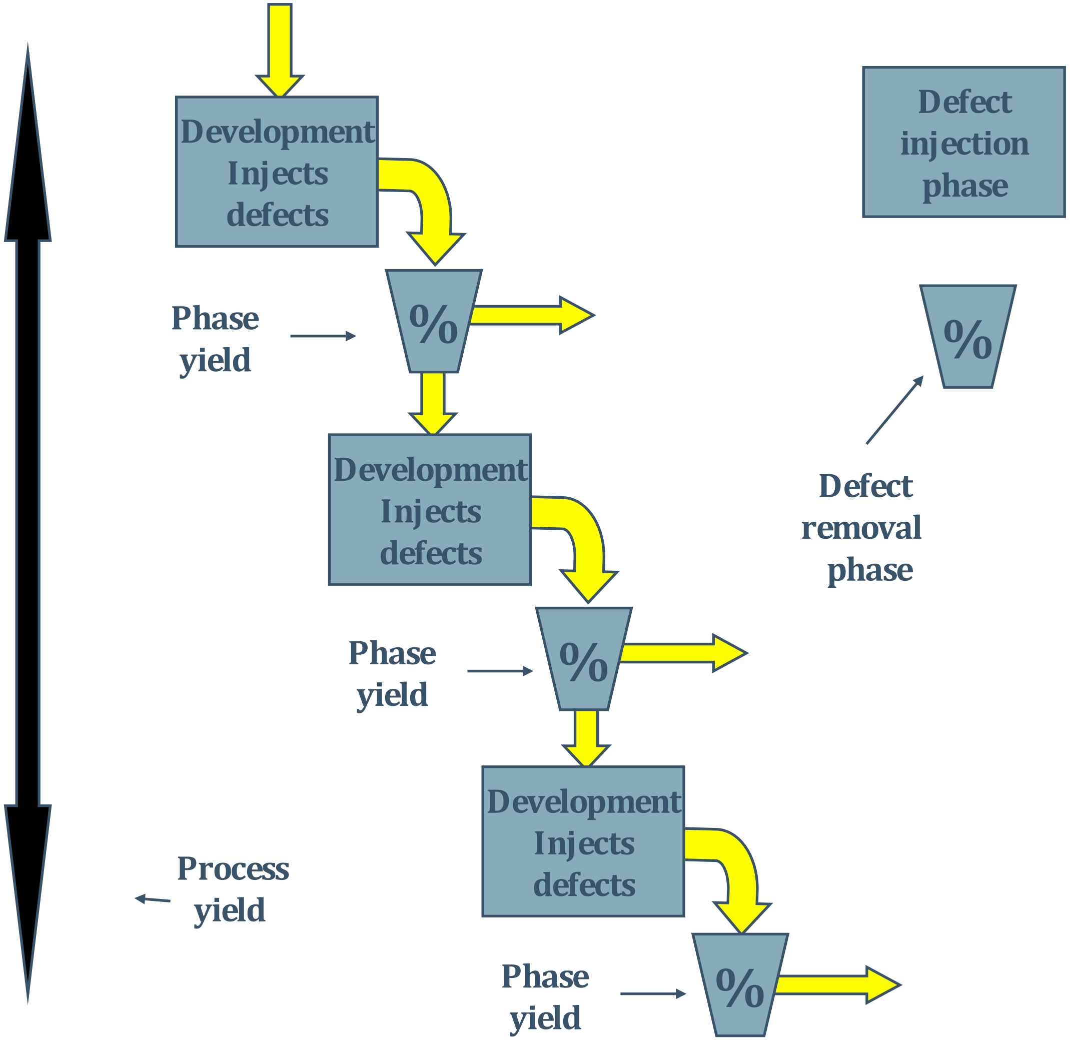
# Phase Yield

Phase yield measures the
percentage of the defects in the product that were found by that phase
defect-removal effectiveness of that process step

Yield can be used to measure the effectiveness of design and code reviews, inspections, compiling, and testing.

Yield (for a phase) =
100 * (defects found) / (defects found + not found)

# Defect-Removal Filters



# Yield Estimates

Yield can be estimated but not precisely calculated until all defects have been found through test and product use.

Yield measures are most useful when the developers and testers record all of the defects.
design and code review defects
compile defects
test defects

By using process-control measures, you are more likely to do high-yield reviews.

# Potential Control Parameters

To be useful, process control measures must be available during the process. Examples are
size units reviewed per hour
defects found per hour
defects found per size unit

While no control parameter directly correlates with phase yield, review rate is the most useful control parameter.

Review rate is the parameter used in the PSP.

# TSP Overview

## Leading a Development Team

# What Is the Team Software Process?

The Team Software Process (TSP) is an integrated set of  practices for developing software.

The TSP is a process-based solution to common software engineering and management challenges:

- cost and schedule predictability
- productivity and product quality
- process improvement

# Topics

The TSP

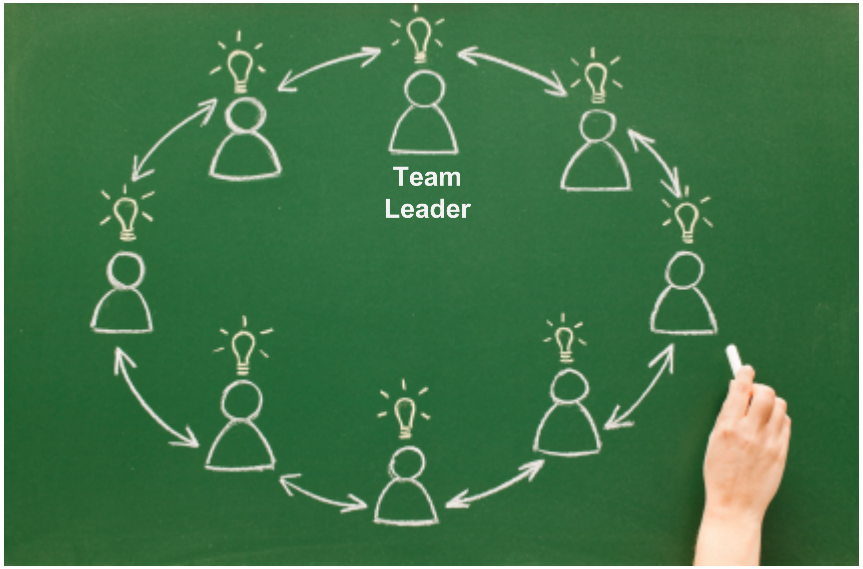
- establishes self-managed teams
- is guided by a defined process framework
- provides a team-based project planning process guided by a coach
- includes metrics for tracking project status and product quality
- provides immediate, measurable benefits

# TSP Team Member Roles

All team members have traditional roles; for example developer, tester, or inspector.

They also share self-management responsibilities through eight defined management roles.

- Customer Interface
- Design
- Implementation
- Planning
- Process
- Quality
- Support
- Test



# The Team Leader’s Role

The team leader does not typically take one of the aforementioned eight team roles.

The team leader’s role on a TSP team is to

- guide the team in doing its work
- support the self-managed team concept
- take the time to reach full consensus on all important issues
- ensure that the team establishes high standards for the work
- promote and facilitate team problem solving
- provide management support to the team and interface with management for the team
- protect the team so that team members can focus on the project

# The TSP Defined Process Framework

TSP is based on a defined process framework.

The TSP process framework was designed to be integrated with existing processes, methods, and tools.

Processes within the framework can be tailored by the team under the guidance of a TSP coach.

The TSP process

- guides the work (but doesn’t constrain it)
- ensures consistency of results
- promotes effective communication
- stimulates individual and team improvement

# Algunos Elementos del TSP

- Fases
- Medidas
- Roles
- *Scripts* (guías para expertos)
- *Best practices*
- Herramienta de soporte al proceso

# Fases

El TSP es un marco, por ende, las fases pueden ser ampliadas y cambiadas. Sin embargo, se debe respetar el marco en lo que refiere a las *best practices*.

- Planning
- Requirements
- System Test Plan
- REQ Inspection
- High-Level Design
- Integration Test Plan
- HLD Inspection

# Fases (2)

- Detailed Design
- DLD Review
- DLD Inspection
- Code
- Code Review
- Compile
- Code Inspection
- Unit Test

PSP + Inspecciones

# Fases (3)

- Build and Integration Test
- System Test
- Postmortem

# Medidas

- El TSP recolecta datos de la ejecución del proceso así como del producto.
- Estos datos sirven para planificar (datos históricos) y para hacer el seguimiento del proceso, producto y proyecto.
- Las medidas base son las mismas que en el PSP.

# Scripts y Best practices

- *Scripts*
  - Cada una de las fases del TSP tiene al menos un script asociado. Los scripts, al igual que en el PSP, guían el trabajo del experto.
- *Best practices*
  - Estas son prácticas fundamentales que permiten mantener la calidad del producto y, a través de esta, mantener controlado el proyecto/proceso/producto.
  - Actividades de prevención: planificación, diseño y postmortem.
  - Actividades de remoción temprana de defectos: revisiones e inspecciones.

# Planning and Team Building

The *TSP launch* process addresses planning and team building.

A typical launch takes two-four days. Small teams need less time.

- The launch is considered part of the project work.
- All team members participate.
- The launch is guided by a TSP coach.

Management provides the goals and constraints for the project.

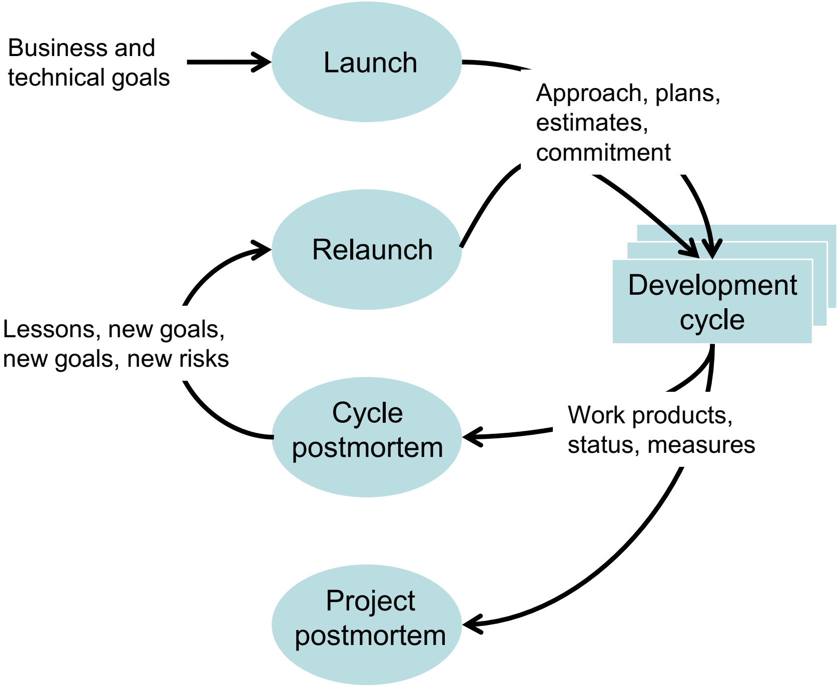
The team then develops its plan for meeting management's objectives.

If necessary, the team negotiates with management to arrive at a mutually agreeable outcome.

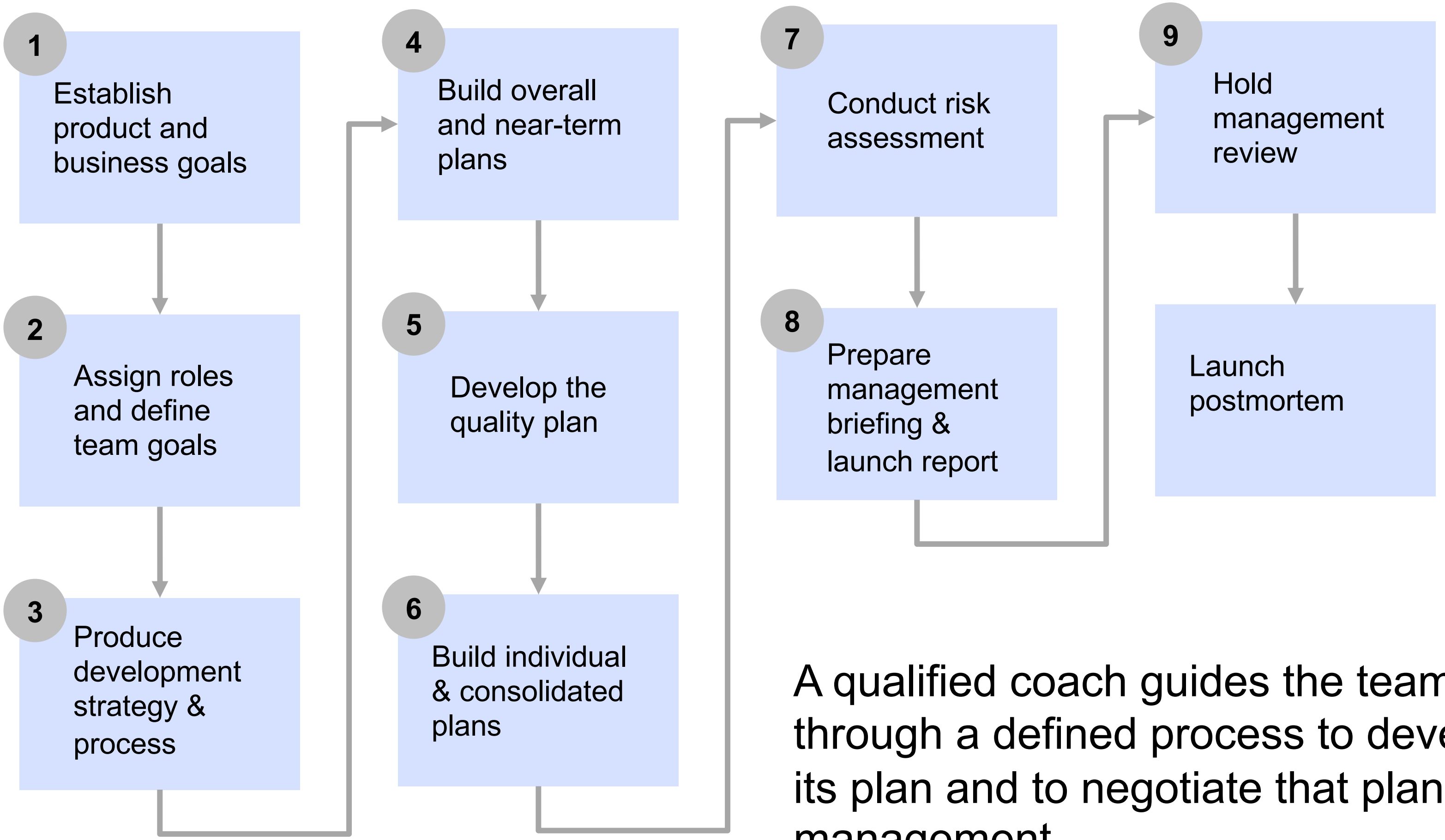
# TSP Development Strategy

A team can begin using TSP on a newly started project or on a project that is already underway (any cycle or phase of a cycle).

The TSP planning cycle begins with a launch or a re-launch and ends with a postmortem.


Business and technical goals
Launch
Approach, plans, estimates, commitment
Relaunch
Development cycle
Lessons, new goals, new goals, new risks
Cycle postmortem
Work products, status, measures
Project postmortem

# The TSP Launch Process


1
Establish product and business goals
2
Assign roles and define team goals
3
Produce development strategy & process
4
Build overall and near-term plans
5
Develop the quality plan
6
Build individual & consolidated plans
7
Conduct risk assessment
8
Prepare management briefing & launch report
9
Hold management review
Launch postmortem

A qualified coach guides the team through a defined process to develop its plan and to negotiate that plan with management.

# Class Discussion - Coaches

What is the role of a coach on a team?

# The TSP Coach

The TSP coach is a TSP expert who works collaboratively with the team leader to ensure project success. The coach

- guides the TSP team through the TSP launch process
- coaches the TSP team members as they work on the project
- is not a team member or the team leader

The TSP coach is an expert in the TSP process.

- TSP information products provide expert-level guidance with the assumption that a coach will be assisting the team.
- Every team and project is different and how the TSP is specifically implemented depends on the particular project context.  The coach assists the team in tailoring the TSP appropriately.

# Tracking and Status Reporting

A self-managed team makes their own plan and is held accountable for managing their plan.

For a TSP project, four base measures are collected to support tracking against the plan. They are:

- schedule (that is, completion status of tasks)
- size
- time
- defects

# What the Base Measures Provide

Base Measures provide the necessary information to answer important questions about the project.

- When do we expect to finish?
- How confident are we in that expectation?
- Why are we behind/ahead of schedule?
- How do we spend our time?
- What do we expect the quality of the final product to be?
- How confident are we in that expectation?
- When are defects being injected and removed?
- How effective are our reviews and inspections?

# TSP Benefit: Reduction in System Test Time

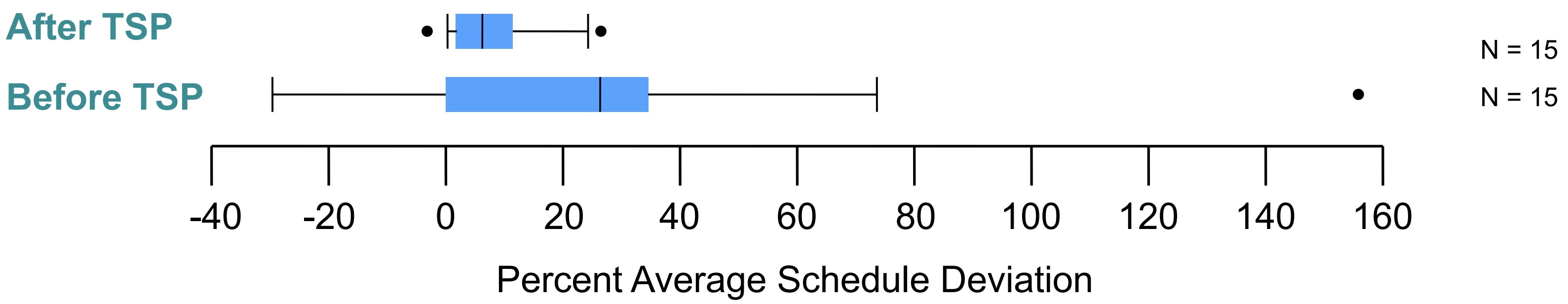
Typical software project teams spend 40-50% of their effort in system test, finding and fixing defects.

By sending cleaner code to system test, the total time spent in system test can be reduced to 9-15%.

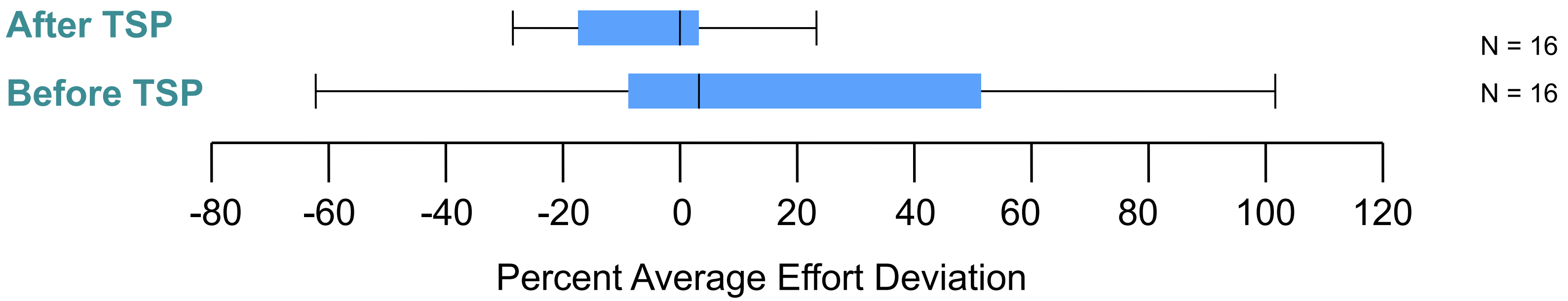
Reducing system test time increases the time that teams can spend in developing product during a typical year-long release cycle … by almost 50%.

# Results: Schedule and Effort Deviations

## Schedule Deviation



## Effort Deviation

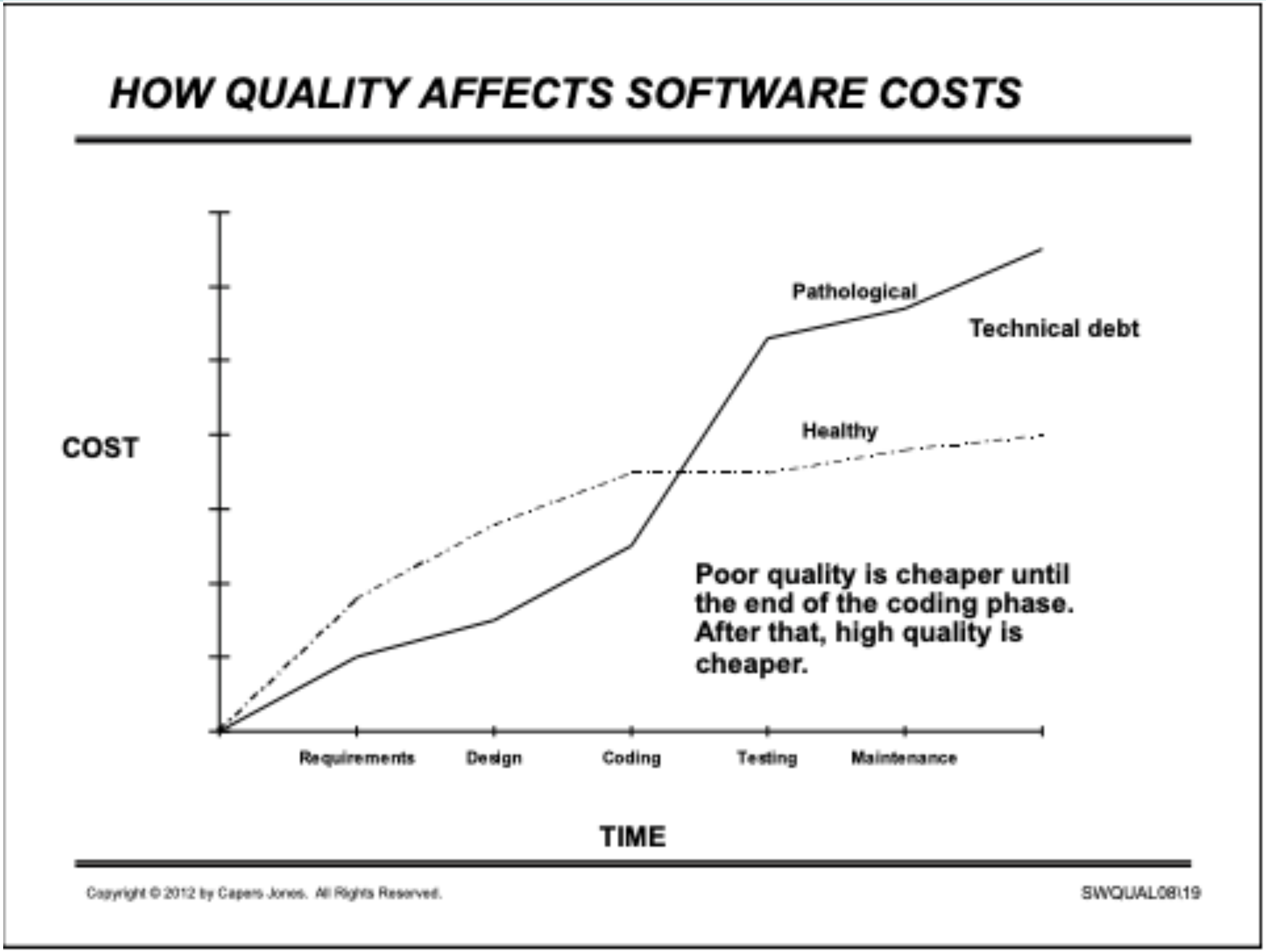


🟦 25th to 75th percentile

├ Max/min data point

● Outlier

[Source: McAndrews, D. 2000]



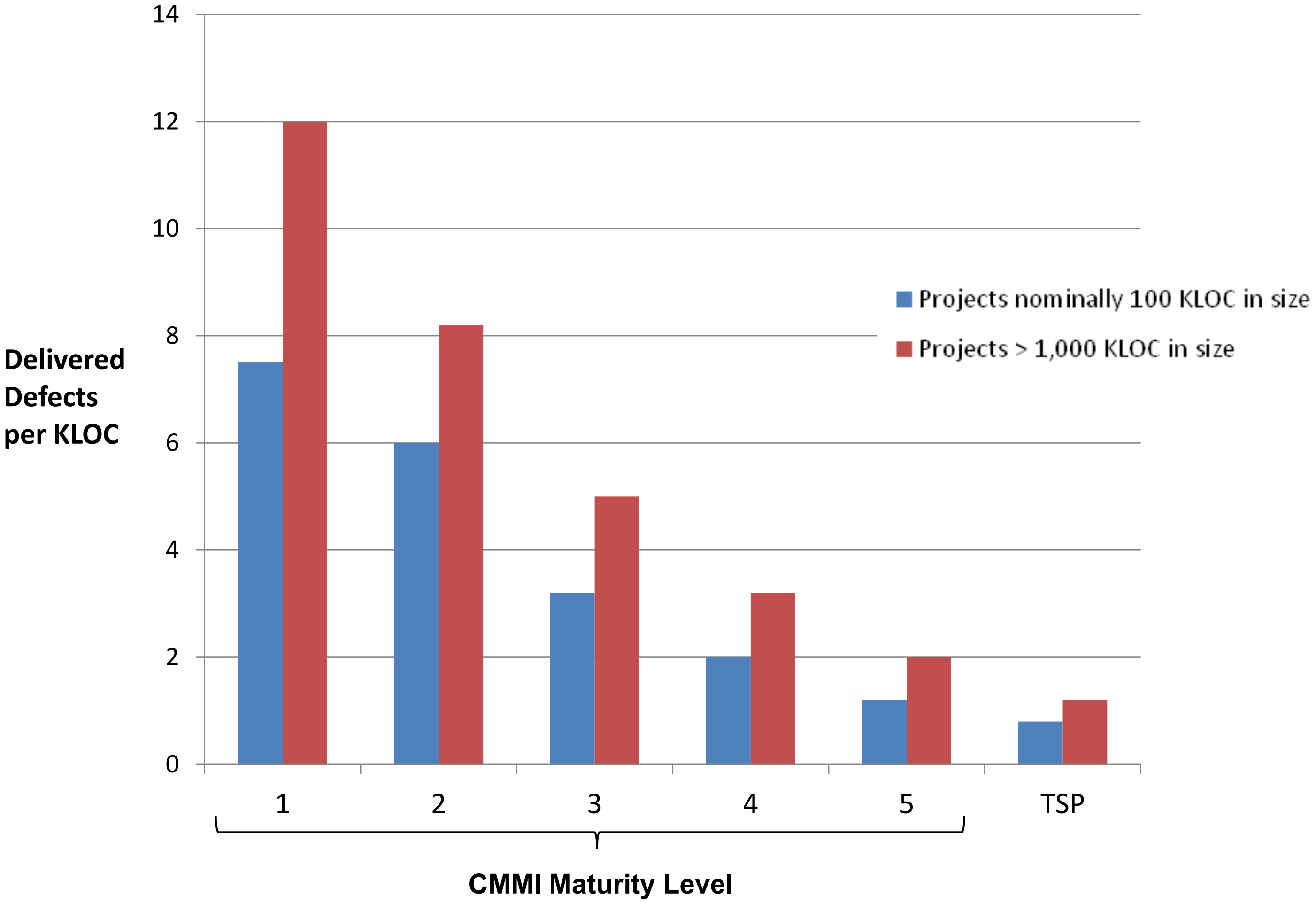
## HOW QUALITY AFFECTS SOFTWARE COSTS

Poor quality is cheaper until the end of the coding phase. After that, high quality is cheaper.

SWQUAL08\19

“It is an interesting and important point that projects with low defects potentials and high defect removal efficiency (DRE) above 96% percent are faster and cheaper than the same size projects with poor DRE below 90%.” Capers Jones

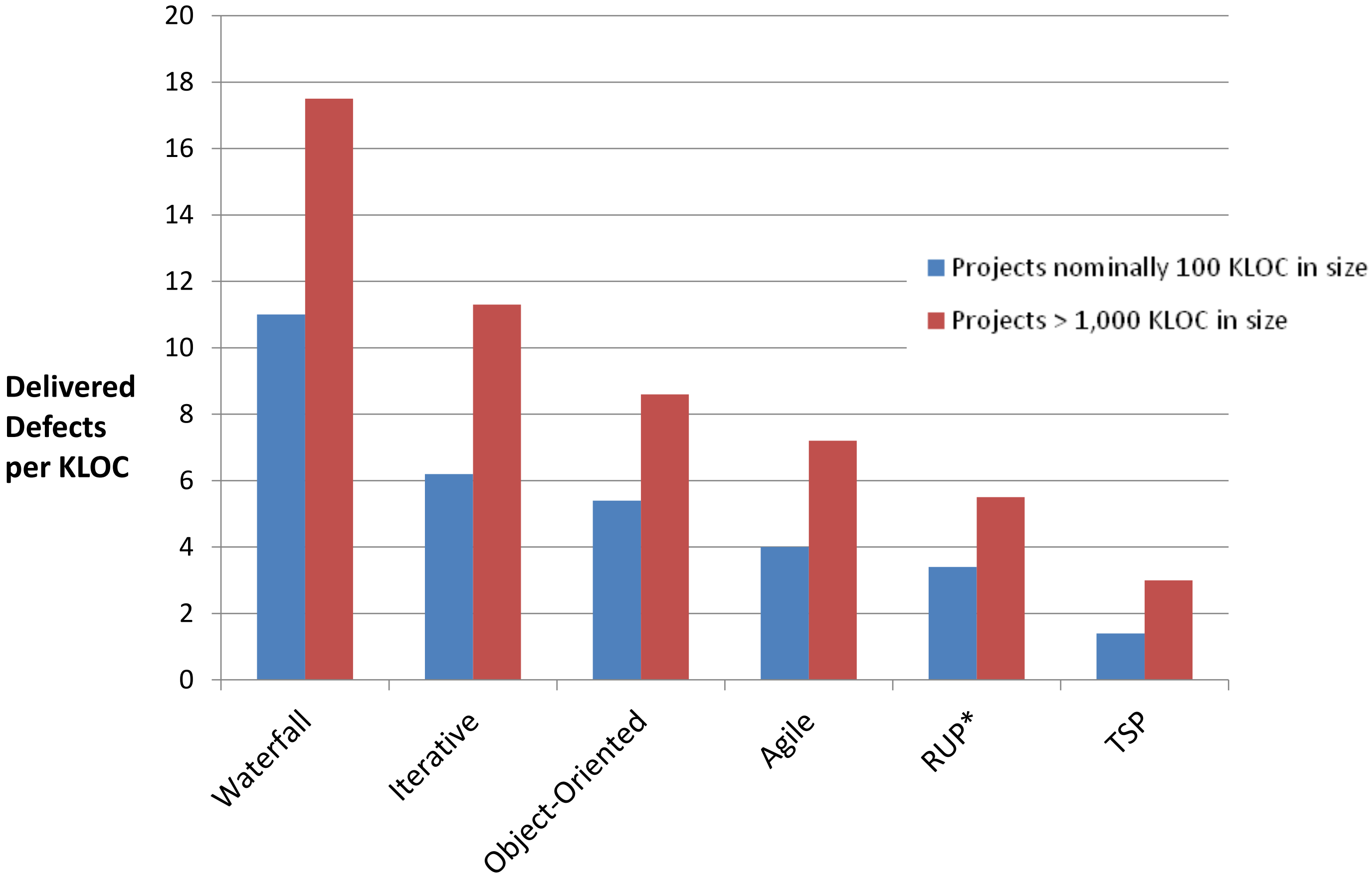
# Results: Defects Delivered



Note: Function points were translated to KLOC using backfire method (1FP = 100 LOC) .

[Source: Jones, C. 2011]

# TSP Compared to Other Methodologies



Note: Function points were translated to KLOC using backfire method (1FP = 100 LOC) .

* RUP is Rational Unified Process.

[Source: Jones, C. 2012]

# The Importance of Developer Training

In order to be successful with TSP, each member of the team is trained so that they acquire skills to

- make realistic plans
- negotiate commitments
- track their work
- produce high quality products

Training

developing the skills,
employees need to perform the
improve their performance kn
skills, and abilities, specific tc

# References

McAndrews, D. *The Team Software Process: An Overview and Preliminary Results of Using Disciplined Practices* (CMU/SEI-2000-TR-015, ADA387260). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000. http://www.sei.cmu.edu/publications/documents/00.reports/00tr015.html.

Jones, C. *Software Quality in 2010: A Survey of the State of the Art.* http://www.sqgne.org/presentations/2010-11/Jones-Nov-2010.pdf (2011).

Jones, C. *Software Quality in 2011: A Survey of the State of the Art.* http://sqgne.org/presentations/2011-12/Jones-Sep-2011.pdf (2012).