

Watts Humphrey's PSP

- **Personal software process**
- **Scales down CMM practices**
- **Builds individual capabilities**
- **Supports organization processes**

A Discipline for Software Engineering,
Addison-Wesley, 1995.

PSP Course Approach

- **Series of programs to write**
 - From simple to more complex
 - Programs a vehicle, not an end
- **Series of software processes**
 - Start with basics
 - Add elements at each level

Baseline Process (PSP0)

- **PSP0**
 - Time & defect recording
 - Project plan summary
 - Standard forms
 - Specific procedures (scripts)
- **PSP0.1**
 - Add size "guesstimate", measurement
 - Coding standard, PIP (process improvement proposal)

PSP1 Processes

- **Elements from PSP0.1**
- **PSP1 adds:**
 - Size estimation with PROBE
 - Formal test and test reporting
- **PSP1.1 adds:**
 - Time estimation with PROBE
 - Task and schedule planning

PSP2 Processes

- **Elements from PSP1.1**
- **PSP2 adds:**
 - PROBE prediction interval
 - Defect estimation
 - Design/code reviews, yield tracking
- **PSP2.1 adds:**
 - Formal design templates
 - Cost of quality: appraisal, failure, A/F

PSP3 Process

- **Scale PSP2.1 up**
 - To handle larger programs
- **Subdivide total development**
- **Overall planning & postmortem**
- **Multiple development cycles**
 - Each cycle uses PSP2.1 process
- **Avoid exponential growth??**

TSP Process

- **Team Software Process**
- **Team roles**
- **Statistical process control**
- **“Level 5 process for teams”**
- **New Humphrey book in progress**

Planning

- **Explicit statement of work**
- **Project task breakdown**
- **Estimates**
 - Base on prior work
 - Record and review versus actual results

Size Measurement

- **LOC counting standards**
 - Count comments?
 - Count lines or statements?
- **Coding standards**
 - Standardize coding format
 - Simplify counting LOC's

Size Estimation

- **Proxy-based estimating (PROBE)**
 - E.g., house square footage predicts building costs
- **Possible proxy**
 - Object lines of code
 - # of lines for small, medium, large ...

Resource/Schedule

- **Estimating development time**
 - Correlate object LOC and hours
 - Based on historical data
- **Estimating other task time**
 - Reports, etc.
 - Establish process measures

Earned Value Tracking

- **Assign value to each task**
 - As percent of overall project
- **Credit for completed tasks**
 - Even if not done in planned order
- **Track for project management**

Reviews

- **Design and code reviews**
- **Checklists**
- **Review against coding standards**
- **Reviews vs inspections (?)**
 - Personal versus team review

Software Quality

- **Economics of quality**
 - Cost to find/fix defects
 - Economics of defect removal
 - By stage of development process
- **Cost of quality**
 - Yield management
 - Defect prevention

Software Design

- **Design process structure**
 - Requirements, specification, etc.
- **Design notation**
 - Functional specification template
 - Formal notation for object methods
 - Pre-conditions and actions
 - Like Rose: pre, semantics, post

State Specification

- **Internal object dynamics**
 - Defined states
 - Transition conditions
 - Map all pairs of states
 - Valid conditions or “impossible”
- **Graphical representations**
 - Like Harel state chart

Logical Specifications

- **For each object method (function)**
- **Standard template**
 - Includes
 - Type definitions
 - Declarations
 - Program logic (pseudocode?)

Operational Scenario

- **Like UML sequence diagram**
 - Use case, step-by-step description
- **Standard template**
 - Visualize behavior
 - Document exception conditions

Verification

- Loops
- State machines
- Proof by induction
- Execution and trace tables
- Formal verification

Software Process

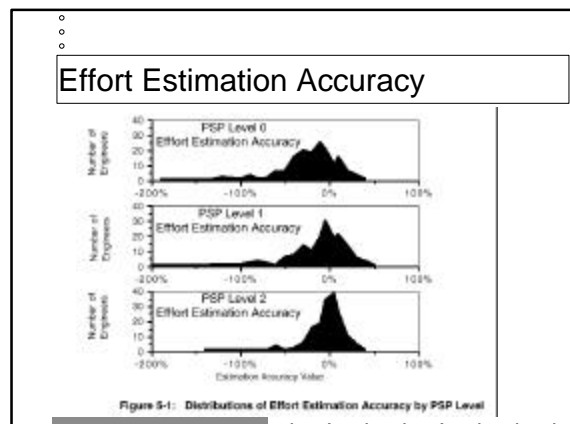
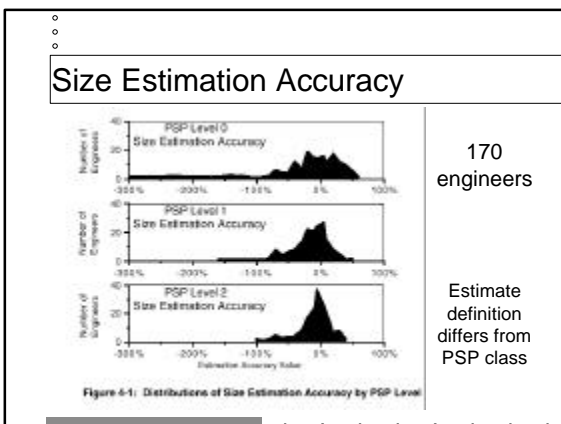
- Process elements
 - Scripts, forms, standards
 - Improvement proposals
- Needs and priorities
 - Quality Function Deployment (QFD)
 - Objectives, goals, criteria

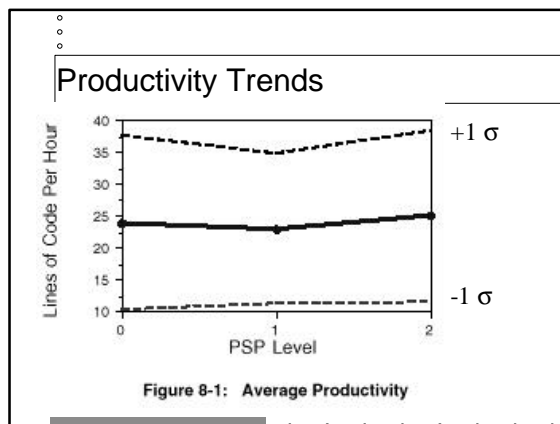
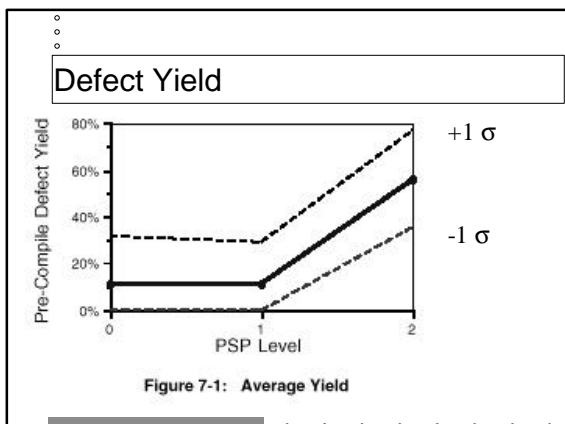
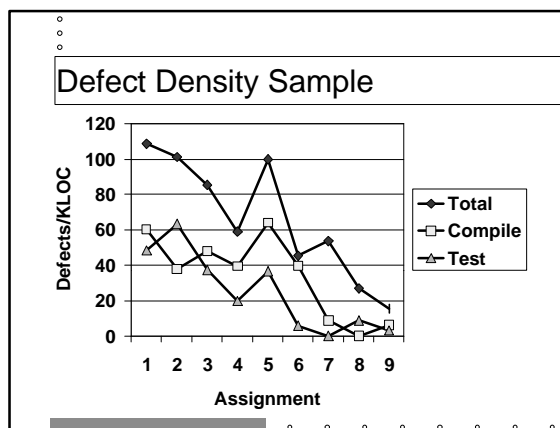
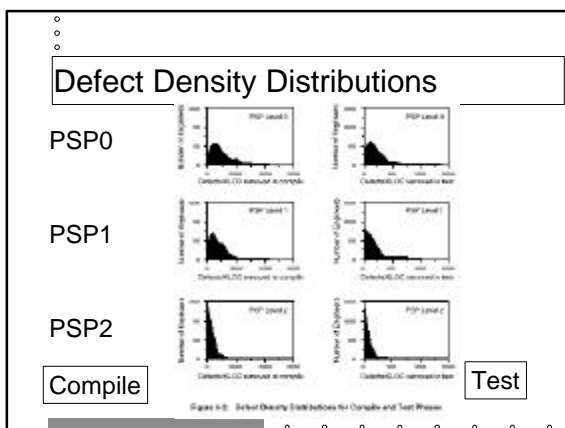
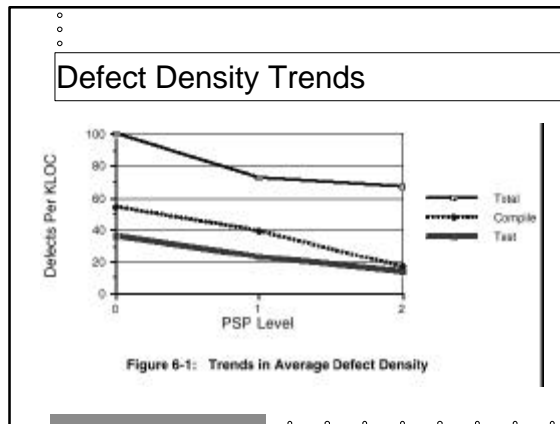
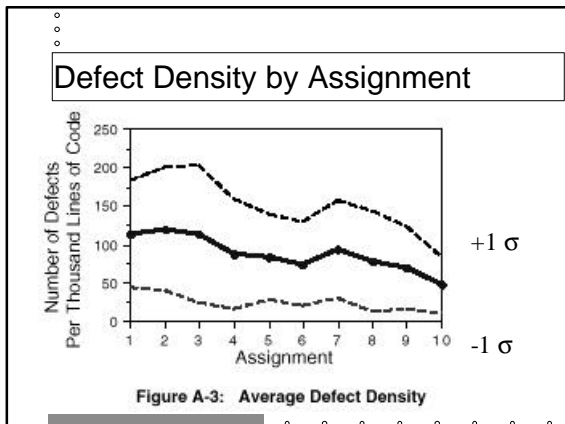
Software Process

- Characterize current process
- Characterize target process
- Establish process development strategy
 - Define process
 - Validate initial process
 - Evolve process

Course Results

- Data for 298 engineers (or subset)
 - Total of 23 PSP courses
 - CMU/SEI=97-TR-001
- The data are on:
 - Estimation accuracy
 - Defect density
 - Productivity





Instructor Data

- **Compressed PSP workshop**
 - Three programs (process PSP0, 1, 2)
 - Third one an interesting story?
- **One post-workshop program**
 - Program information (LOC counting!)
 - Like project phase 1
 - Rose OOA/OOD, MFC, etc.

