 **Software Engineering Institute** | Carnegie Mellon

PSP Advanced

Course Overview

Sponsored by the U.S. Department of Defense
© 2010 by Carnegie Mellon University
This material is approved for public release. Distribution is limited by the Software Engineering Institute to attendees.

PSP Advanced: Course Overview June 2010

Lecture Topics

Introductions

PSP Fundamentals Review

Course Objectives

Course Agendas



PSP Advanced: Course Overview

Introductions

Instructor introductions

Building facilities

Student introductions

- name and organization
- course expectations
- PSP and TSP experience
- programming language you will be using

© 2010 Carnegie Mellon University 3

Set the example by keeping your instructor introductions short and to the point. Emphasize that they do the same, so as to move beyond this point quickly.

As students articulate their course expectations. Note them on a board or flip chart so that you can come back to them at the end of the course. Make sure to note which expectations will and will not be covered as the students introduce themselves in order to set realistic expectations.

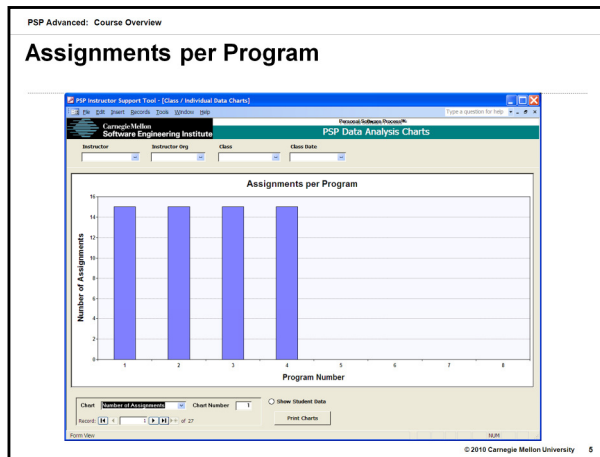


What did we learn in PSP Fundamentals?

- What is a Personal Software Process
- How to use PSP Forms, Scripts, Standards, tools, etc
- How to collect and use PSP core measures:
 - effort (time)
 - size (LOC, pages, etc.)
 - schedule
 - quality (defects)
- How to make size and effort estimates
- How to apply different defect removal techniques
- How to perform schedule planning and tracking

Briefly recap the material covered in the PSP Fundamentals course.



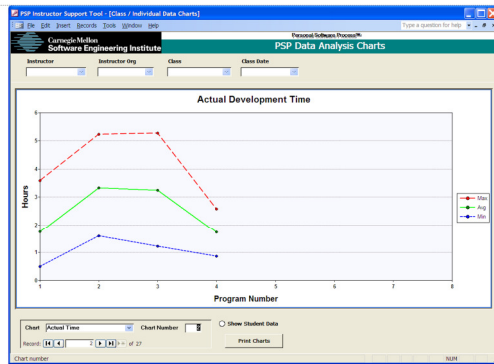


Instructor's choice. The following graphs should be covered as a review of the PSP Fundamentals Course. It represents real data from 15 students that have taken the PSP Fundamentals Course. Instead of going over these graph slides. The instructor could collect the PSP Advanced student's fundamental data prior to the first day of class and present the student's class composite data using the PSP Instructor Support Tool.

NOTE: The instructor needs to go through these graphs quickly, in order to stay on schedule. It is only a review of what the students should have already learned in the PSP Fundamentals course.



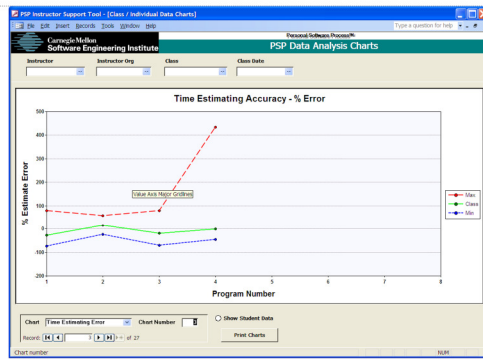
Actual Development Time



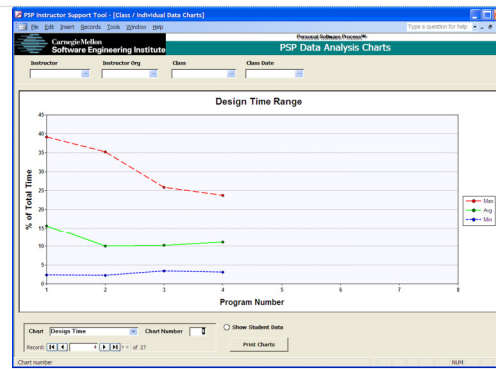
© 2010 Carnegie Mellon University 6



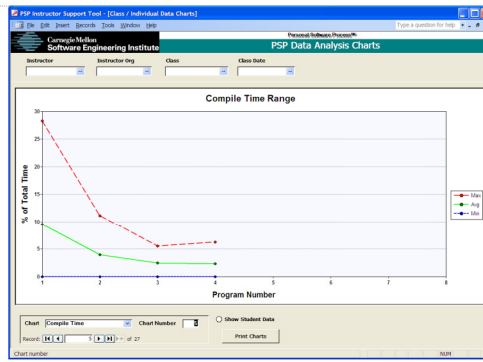
Time Estimating Accuracy - % Error



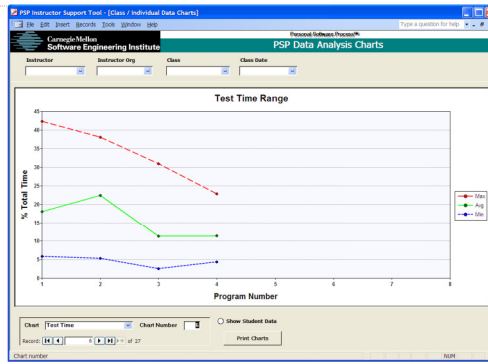
Design Time Range



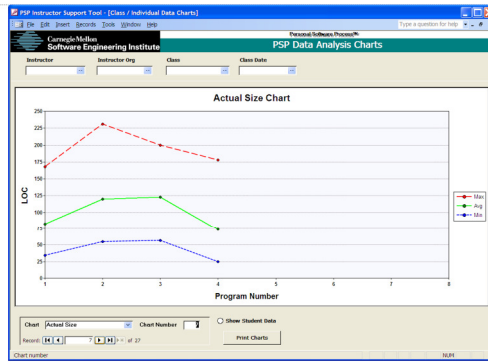
Compile Time Range



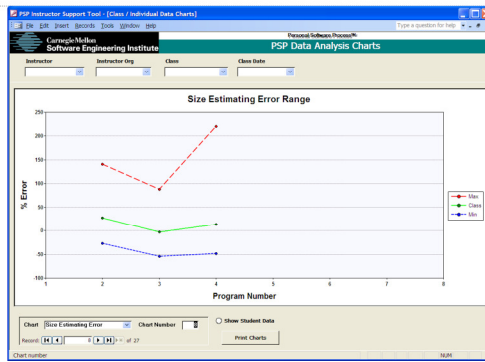
Test Time Range



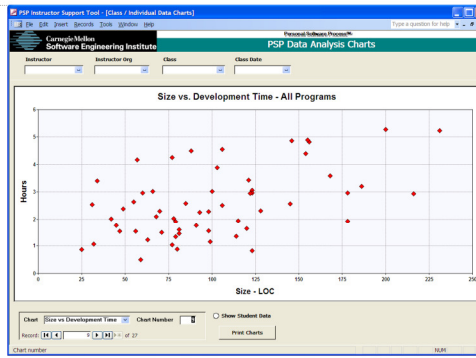
Actual Size Chart



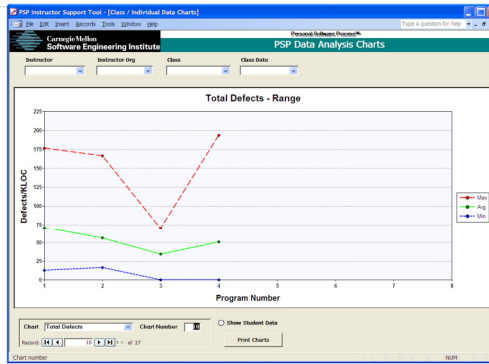
Size Estimating Error Range



Size vs. Development Time – All Programs



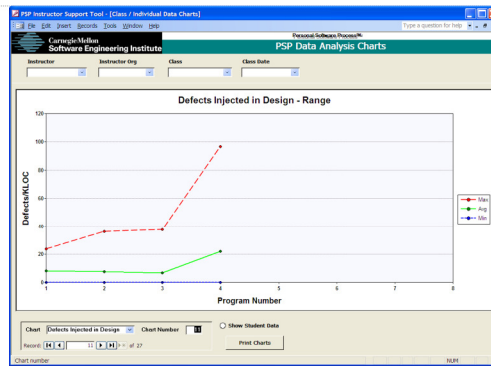
Total Defects - Range



© 2010 Carnegie Mellon University 14



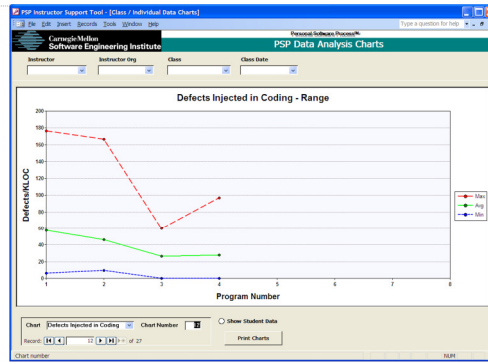
Defects Injected in Design - Range



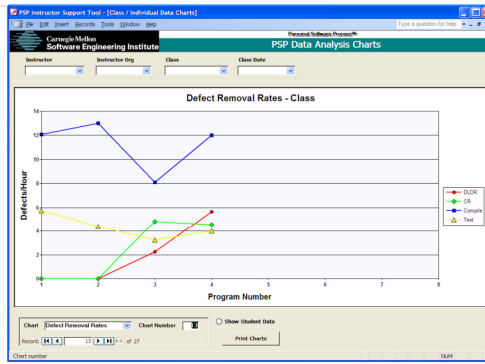
© 2010 Carnegie Mellon University 15



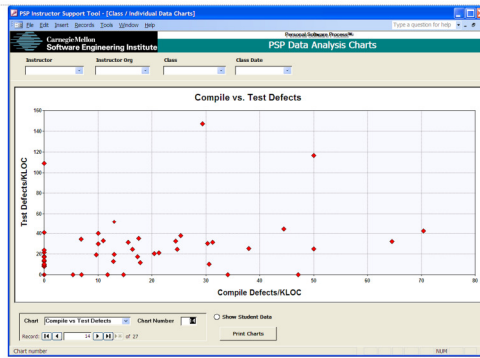
Defects Injected in Coding - Range



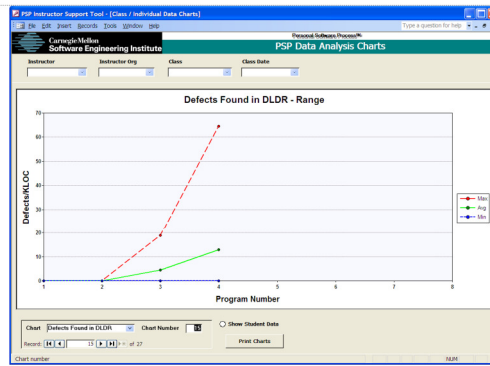
Defect Removal Rates - Class



Compile vs. Test Defects



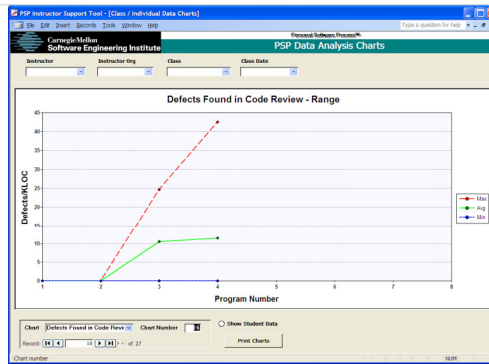
Defects Found in DLDR - Range



© 2010 Carnegie Mellon University 19



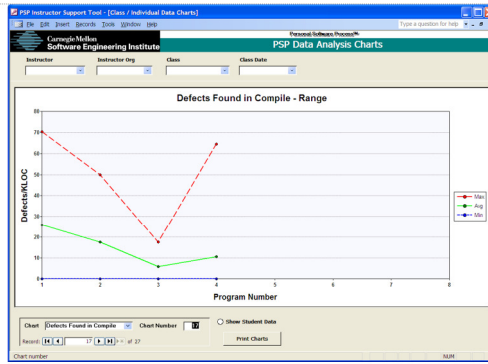
Defects Found in Code Review - Range



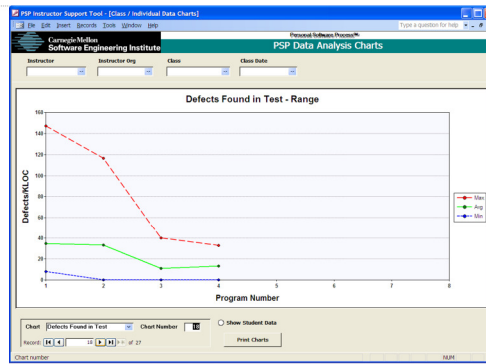
© 2010 Carnegie Mellon University 20



Defects Found in Compile - Range



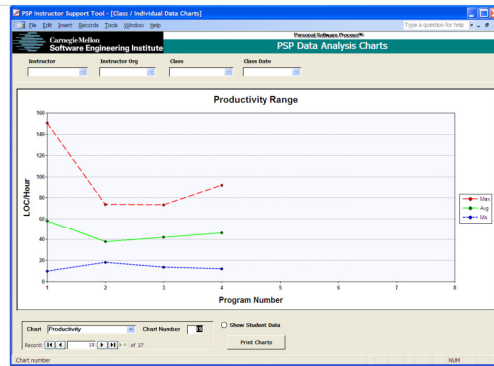
Defects Found in Test - Range



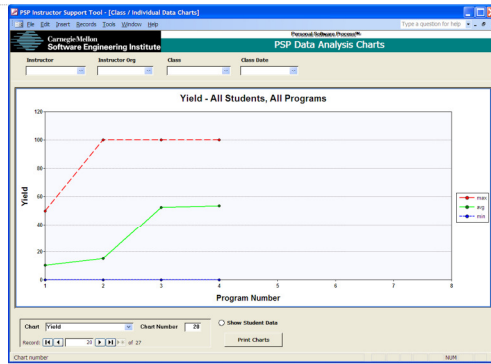
© 2010 Carnegie Mellon University 22



Productivity Range



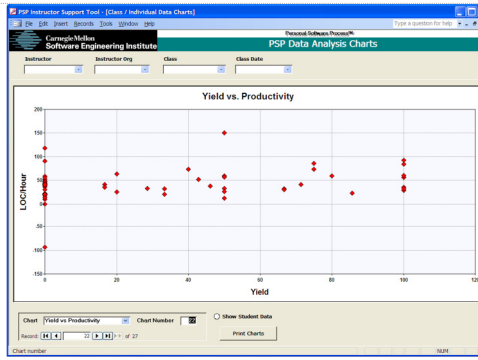
Yield – All Students, All Programs



© 2010 Carnegie Mellon University 24



Yield vs. Productivity



PSP Advanced Course Objectives

Provide you with the skills and methods to

- analyze your current performance
- improve your current performance based on the analysis by making process improvements
- extend the PSP Fundamentals methods with more best practices



Course Structure

Pre-course reading assignment

Typical course day

- lecture
- lab
 - process tutorial
 - programming or report assignment
 - lab work



Instructor Availability

Instructors will be available

- during class hours each day
- after class by phone or e-mail



Ground Rules for Class

Be here on time; we will start on time each day.

Attend all sessions; discuss any exceptions with instructors.

Keep this room a "quiet zone" during lab periods; move conversations outside.

You must use the PSP techniques/methods on the assignments. Your goal is to learn the process, not just to write programs.

Turn in assignments that are correct and complete.

Participate! This course is as good as the discussions we will all have together.



Course Agenda - Day 1

- 8:00 Continental breakfast
- 8:30 L1: Course Overview
- 9:15 [L2: Software Design](#)
- 10:15 Break
- 10:30 [Design Specification Templates Exercise](#)
- 11:15 [L3: Design Verification](#)
- 12:00 Lunch
- 1:00 [Design Verification Exercise](#)
- 1:30 [PSP2.1 Tutorial](#)
- 1:45 Lab session
 - [Program 5 Assignment](#)



Course Agenda - Day 2

- 8:00 Continental breakfast
- 8:15 [Class data feedback](#)
- 9:00 [L4: Planning and Tracking Commitments](#)
- 10:15 Break
- 10:30 [L5: Understanding and Improving Planning Performance](#)
- 11:15 [PROBE Methods A & B Tutorial](#)
- 12:00 Lunch
- 1:00 Lab session
 - [Program 6 Assignment](#)



Course Agenda - Day 3

- 8:00 Continental breakfast
- 8:15 [Class data feedback](#)
- 9:00 [L6: Understanding and Improving Quality Performance](#)
- 10:15 Break
- 10:30 [L7: Planning and Tracking Quality](#)
- 11:30 Lab session
 - [Program 7 Assignment](#)
- 12:00 Lunch
- 1:00 Lab session (continued)



Course Agenda - Day 4

- 8:00 Continental breakfast
- 8:15 [Class data feedback](#)
- 9:00 [L8: Defining and Improving Personal Processes](#)
- 10:00 Break
- 10:15 [Process Definition Exercise](#)
- 10:45 [Process Definition Tool Tutorial](#)
- 11:15 Process Definition Tool Exercise
- 11:45 Lab session
 - [Begin Performance Analysis Report](#)
- 12:00 Lunch
- 1:00 Lab session (continued)



Course Agenda - Day 5

- 8:00 Continental breakfast
- 8:15 [Class data feedback](#)
- 9:00 [L9: Being a TSP Team Member](#)
- 9:45 Break
- 10:00 [Capstone exercise](#)
- 11:45 [L10: Course Conclusion](#)
- 12:00 Lunch



