Course Goals

After completing this course, you will:

- Have practiced and know how the P[SE]PMP provides a basis for the Project Management quality and productivity improvements the students invariably experience.
- See how PPMP is a basis for on-going improvements and for PSP.
- Have practiced and built personal tools for
  - Earned Value Planning and Tracking
  - PROBE (PROxy Based Estimating)
  - Multiple regression
- Have learned and practiced PSP on a product software engineering task [If you are taking the full CS599]
PSP Course Objectives

To introduce you to a process-based approach to developing software

To show you how to measure and analyze your personal software process (PSP)

To show you how to use process data to improve your personal performance

To show you how to apply these methods to your other tasks

Presentation Outline

Today’s Meeting
- Part 1: Overview and PSP0s
- BREAK (before 11:00)
- Part 2: PPMP Exercise: "Programming" Assignments & Reading
- BREAK (at 12:00)
- Part 3: CS599 Project and Assignments [Separate Material]
- BREAK--Resume next session

Next Meeting
- Part 4: Discuss PPMP Exercise, Demonstration & Reading
- BREAK
- Part 5: More on PPMP and PSP
- BREAK
- Part 3: CS599 Project and Assignments [Separate Material]
Presentation Outline Detail

Part 1: Overview and PSP0s
- Describe each of the PPMP/PSP versions, and why they exist
- Explain PPMP0s' tasks, activities, processes and artifacts
- Describe PPMP/PSP1s' approaches, concepts and techniques, including size estimating approaches and historical databases

BREAK (before 11:00)

Part 2: PPMP Exercise: "Programming" Assignments & Reading
- Demonstrate: "Spreadsheet Programming" & functions
- Practice PSSP0 planning and data gathering

BREAK (at 12:00)

Part 3: CS599 Project & Assignments [Separate Material]
- Projects
- Demonstrate COCOMO II

BREAK--Resume next session

Presentation Outline Detail (cont.)

Next Meeting

Part 4: Discuss PPMP Exercise, Demonstration & Reading

BREAK

Part 5: More on PPMP and PSP
- Describe rest of PPMP's Methods, Techniques & Technologies
- PPMP + PSP
- Personal Quality Management – PSP2s
- Cyclic Personal Process – PSP3
- Personal Process Definition & Development

BREAK

Part 6: CS599 Project and Assignments [Separate Material]
- System Testing a la IEEE STD J-016
- Using UML as source models for testing
Session 1 - Outline

Introductions

Overview of PPMP/PSP

PPMP0s' tasks, activities, processes and artifacts

First Exercise: Simple size data calculations

PPMP process versions

PSP's higher process versions

Introductions

Fill in and hand in ASAP the Student Information Form; Remote students: Please download from http://sunset.usc.edu/classes/cs599s99/studinfo.html & FAX to 213/740-4927 or Email to AWBrown@USC.edu

• Name, Daytime phone number, Evening phone number, ...
• Background

Participants (around room; remote students phone in)

• Name
• Company/Organization & Position
• Reason for interest in PPMP
• Personal goals for course
Course Content

What do you know about PSP?

Any questions?

Course Content – Questions for You

What method(s) do you use for defining work processes?

What kind of "model" of process improvement is the CMM?
   e.g. Staged or Continuous Process Improvement?

What is the CMM? Why does it matter to this course?

Write down a one sentence definition for each of the following and then rank the importance of them in your organization (1 being the highest):

• Quality
• Cost
• Schedule
• Productivity
Introductions

Presenter:

A. Winsor Brown
Assist. Director, USC Center for Software Engineering

– awbrown@sunset.usc.edu
– 213/740-6599

The "Personal Software Process" PSP and PPMP

Overview

A process for individuals to use
Applies to structured personal tasks
• Writing small programs or documents
• Defining requirements or processes
• Conducting reviews or tests, etc.

PPMP Introduced
• In steps; only part of PSP
• With small spreadsheet and writing [nee software] tasks

Is a discipline for personal software engineering project management
The "Personal Software Process" PSP

Strategy

Current state

- Today, software engineers (adapted from W. Humphrey):
  - Learn software development by practicing on classroom ("toy") problems; Graded on what produced
    - Develop their own processes for these toy problems
      - Not graded on how solution is produced (process)
    - Toy processes typically unsuitable as foundation for large-scale software development
  - Learn project "planning and control" as part of a team by practicing on semester length projects (if they are lucky)
    - Develop own processes;
      - Seldom graded on how produced
    - Seldom practice process improvement

Watts' PSP approach

- Start with large system practices
- Select those that are suitable for use by individuals
- Introduce them in defined increments
- Software engineers practice the processes
- Software engineers personally experience their benefits

PPMP approach

- Modify PSP practices for spreadsheets and writing plans
- PPMP practices introduced in defined increments
- PSP practices covered also for 3GL project work
PSP and PPMP learning considerations

- To consistently use effective methods, engineers must believe they are effective
- To believe methods are effective, engineers must use them
- If they do not learn them in school, they must learn them on the job
- Only a few industrial organizations provide a suitable learning environment
The "Personal SE Project Management Process" (PSP) vs. Capability Maturity Model (CMM)

**PPMP & PSP vs. CMM**

**PPMP & PSP** with Key Process Areas

1. **Level 1--Initial**
   - Requirements management
   - Software project planning
   - Software project tracking and oversight
   - Software subcontract management
   - Software quality assurance
   - Software configuration management

2. **Level 2--Repeatable**
   - Software configuration management
   - Software quality assurance
   - Software subcontract management
   - Software project tracking and oversight
   - Software project planning
   - Requirements management

3. **Level 3--Defined**
   - Peer reviews
   - Intergroup coordination
   - Software product engineering
   - Integrated software management
   - Training program
   - Software process definition
   - Software process focus

4. **Level 4--Managed**
   - Software product engineering
   - Integrated software management
   - Training program
   - Organization process definition
   - Organization process focus

5. **Level 5--Optimizing**
   - Process change management
   - Technology change management
   - Defect prevention

**CMM**

1. **Level 1--Initial**
2. **Level 2--Repeatable**
3. **Level 3--Defined**
4. **Level 4--Managed**
5. **Level 5--Optimizing**

**PSP and CMM: Level 2 and 3**

**PSP/PPMP with Key Process Areas**

1. **Level 1--Initial**
2. **Level 2--Repeatable**
3. **Level 3--Defined**

**CMM**

1. **Level 1**
2. **Level 2--Repeatable**
3. **Level 3--Defined**
4. **Level 4--Managed**
5. **Level 5--Optimizing**
PSP and CMM: Level 4 and 5

Level 5--Optimizing:
- Process change management*
- Technology change management*
- Defect prevention*

Level 4--Managed:
- Quality management*
- Quantitative process management*

PSP Phases and Methods

Cyclic Personal Process

Personal Quality Management

Personal Planning Process

PSP0
- Current process
- Time recording
- Defect recording
- Defect type standard

PSP1
- Size estimating
- Test report

PSP1.1
- Task planning
- Schedule planning

PSP2
- Code reviews
- Design reviews

PSP2.1
- Design templates

PSP3
- Cyclic development

PSP0.1
- Coding standard
- Size measurement
- Process improvement proposal (PIP)
PSP/PPMP Steps – Level 0 & 1

Personal Planning Process

PSP1
Size estimating
Test report

PSP1.1
Task planning
Schedule planning

Personal Process Baseline

PSP0
Current process
Time recording
Defect recording
Defect type standard

PSP0.1
Coding standard
Size measurement
Process improvement proposal (PIP)

PSP Steps – Level 2 & 3

Cyclic Personal Process

PSP3
Cyclic development

Personal Quality Management

PSP2
Code reviews
Design reviews

PSP2.1
Design templates
PPMP Phases and Methods

PPMP + PSP Integrated Course Offering

PPMP Topics
+ Spreadsheet Exercises

PSP2 & PSP3 Topics

Project 1st Increment
PPMP Planning (Writing+EVP)
PSP Data Gathering & Size Est'g

Project 2nd Increment
PPMP Planning (Writing+EVP)
PSP Practices + Data Gthr'g
PSP0's Tasks Activities Processes & Artifacts

PSSP0s' & PSP0s' Process

PSSP0 & PSP0 Project Plan Summary

PSSP & PSP Time Recording Log

PSSP & PSP Defect Recording Log

PSSP & PSP0s' Process Decomposition

PSSP0's Process Scripts

PSSP0 & PSP0 Process Flow (Framework)
## PSSP0 & PSP0 Project Plan Summary

<table>
<thead>
<tr>
<th>Time in Phase (min.)</th>
<th>Plan</th>
<th>Actual</th>
<th>To Date</th>
<th>To Date %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate/Compile</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Postmortem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## PSSP0 & PSP0 Project Plan Summary (cont.)

<table>
<thead>
<tr>
<th>Defects Injected</th>
<th>Actual</th>
<th>To Date</th>
<th>To Date %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate/Compile</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Development</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Defects Removed</th>
<th>Actual</th>
<th>To Date</th>
<th>To Date %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate/Compile</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Development</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>After Development</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# PSSP & PSP Process Time Recording Log

<table>
<thead>
<tr>
<th>Date</th>
<th>Start</th>
<th>Stop</th>
<th>Interruption Time</th>
<th>Delta Time</th>
<th>Phase</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

# PSSP & PSP Defect Recording Log

<table>
<thead>
<tr>
<th>Date</th>
<th>Number</th>
<th>Type</th>
<th>Inject</th>
<th>Remove</th>
<th>Fix Time</th>
<th>Fix Defect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Defect Types:**
- 30 Build, Package
- 40 Assignment
- 50 Interface
- 60 Checking
- 70 Data
- 80 Function
- 90 System
- 100 Environment

**Description:**

---

**Description:**

---

**Description:**

---
The "Personal Software Process" PSSP & PSP

**PSSP0s' & PSP0s' Process Decomposition**

**Purpose**
To Guide you in developing module-level spreadsheets

**Inputs Required**
- Problem description
- PSSP0 Project Plan Summary form
- Time and Defect Recording Logs
- Defect Type Standard
- Stop watch (optional)

1. **Planning**
- Produce or obtain a requirements statement.
- Estimate the required development time.
- Enter the plan data in the Project Plan Summary form.
- Complete the Time Recording Log.

2. **Development**
- Design the spreadsheet.
- Code (implement) the design.
- Calculate the spreadsheet and fix and log all defects found.
- Test the spreadsheet and fix and log all defects found.
- Complete the Time Recording Log.

3. **Post-mortem**
- Complete the Project Plan Summary form with actual time, defect, and size data

**Exit Criteria**
- A thoroughly tested spreadsheet
- Completed Project Plan Summary with estimated & actual data
- Completed Defect and Time Recordings Logs
### PSSP0 Planning Process Script

<table>
<thead>
<tr>
<th></th>
<th>Purpose</th>
<th>Entry Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To guide the PSSP &amp; PSP planning process</td>
<td>● Problem description  ● PSSP0 (or PSP0) Project Plan Summary form  ● Time Recording Log</td>
</tr>
</tbody>
</table>

#### 1. Spreadsheet Rqts
- Produce or obtain a requirements statement for the spreadsheet
- Ensure the requirements statement is clear and unambiguous.
- Resolve any questions.

#### 2. Estimate Resources
- Make your best estimate of the time required to develop this spreadsheet.

<table>
<thead>
<tr>
<th></th>
<th>Exit Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>● Documented requirements statement  ● Project Plan Summary with estimated development time data  ● Completed Time Recording Log</td>
</tr>
</tbody>
</table>

### PSSP0 Development Process Script

<table>
<thead>
<tr>
<th></th>
<th>Purpose</th>
<th>Entry Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To guide the development of small spreadsheets</td>
<td>● Requirements statement  ● Time and Defect Recording Logs  ● Project Plan Summary with estimated development time  ● Defect Type Standard</td>
</tr>
</tbody>
</table>

#### 1. Design
- Review the requirements and produce a design to meet them.
- Record time in Time Recording Log.

#### 2. Code
- Implement the design: automatic calculation off & formula mode.
- Record in the DRL any requirements or design defects found.
- Record time in Time Recording Log.

#### 3. Calculate
- Calculate the spreadsheet until error free.
- Fix all defects found.
- Record defects in Defect Recording Log.
- Record time in Time Recording Log.

#### 4. Test
- Test until all tests run without error.
- Fix all defects found.
- Record defects in Defect Recording Log.
- Record time in Time Recording Log.

<table>
<thead>
<tr>
<th></th>
<th>Exit Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>● Thoroughly tested spreadsheet  ● Completed Defect Recording Log  ● Completed Time Recording Log</td>
</tr>
</tbody>
</table>
### PSSP0 Postmortem Process Script

<table>
<thead>
<tr>
<th></th>
<th>Purpose</th>
<th>Entry Criteria</th>
</tr>
</thead>
</table>
|   | To guide the PSSP & PSP postmortem process | **Problem description and requirements statement**  
**Project Plan Summary with planned development time**  
**Completed Defect Recording Log**  
**Completed Time Recording Log**  
**A tested and running spreadsheet** |
| 1. | Defects Injected | Determine from the DRL the number of defects injected in each phase. Enter this number under Defects Injected—Actual on the PPS. |
| 2. | Defects Removed | Determine from the Defect Recording Log the number of defects removed in each PSSP0 (or PSP0) phase. Enter this number under Defects Removed—Actual on the Project Plan Summary. |
| 3. | Time | Review the completed Time Recording Log. Enter the total time spent in each PSSP & PSP0 phase under Actual on the Project Plan Summary form. |

<table>
<thead>
<tr>
<th>Exit Criteria</th>
</tr>
</thead>
</table>
| Complete Project Plan Summary form  
Completed Defect and Time Recording Logs |

### Watts' Process Elements

**Software & Project Management Process Elements**

- Defined scripts for performing the tasks
- A set of
  - forms
  - templates
  - instructions
  for
  - planning
  - tracking
  - recording
- An improvement procedure, with process improvement proposal (PIP) forms and facilities
First Exercise — Spreadsheet #1

Standard Deviation Demonstration

Follow Along [all with laptops]

1. Demo 1S References: DSE Appendix A2
2. Spreadsheet Demonstration 1S Requirements

Write a spreadsheet to estimate the mean and standard deviation of a sample of \( n \) real numbers. The mean is the average of numbers. The standard deviation calculation is described in A2 (but you may use built in functions if you wish).

Do a graphic frequency distribution of the 1S data. Indicate on it the mean and +/- 1, 2 and 3 sigma values.

Hints:

- Use the spreadsheet "template"
- Use built-in function StDev for standard deviation

Exercise Appendix: DSE App A2 abbreviated version

\[
Std = \sigma = \sqrt{\frac{\sum_{i=1}^{n} (x_i - x_{avg})^2}{n-1}}
\]

where \( i \) is an index to the numbers, and \( x_{avg} \) is the average value of the numbers.

<table>
<thead>
<tr>
<th>Item Num</th>
<th>Total (LOC)</th>
<th>( (x_i - x_{avg})^2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>186</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>699</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>132</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>272</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>291</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>331</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>199</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1890</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>788</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1601</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Totals</th>
<th>Variance</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Spreadsheet "Operational" Guidance

Spreadsheet "Programming" as PSP learning tool:
simulate the programming experience:

• do "coding"
  – in formula mode or its equivalent
    (Tools/Options/View/Formulas[Window Option])
  – with "no calculation" on entry
    (Tools/Options/Calculation/Manual).
  – on syntax errors, record the time spent correcting as
    PSP "Compile" phase

• do "testing"
  – turn formula mode off
  – "test" your spreadsheet.
  – Test your spreadsheets expandability (adding data)

Spreadsheet "Operational" Guidance (cont.)

Spreadsheet "Programming" (cont.)

• use any applicable built in functions in your spreadsheet
  – specific function hints often be provided

• use graphs where ever appropriate
  – to check your work
  – to validate your data.

• keep things simple:
  – Don't nest functions;
  – transform data from one area to another in simple, logical
    steps;
  – refer back to base input data (use named ranges); etc.
Spreadsheet "Operational" Guidance (cont.)

Spreadsheet "Programming" (cont.)

- exercise descriptions and test data provide an indication of the desired precision of the results
- where data are provided in a tabular format, duplicate that format in the design of your spreadsheet (even though not explicitly call for)

First Exercise — Spreadsheet Demonstration #1

Standard Deviation & Charts

Repeat this demonstration as a learning step

Kit 1S

- Exercise: Standard Deviation Demonstration
- PSSP/PSP0
- Time Recording Log
- Defect Recording Log
# PSSP0 & PSP0 Project Plan Summary

<table>
<thead>
<tr>
<th>Time in Phase (min.)</th>
<th>Plan</th>
<th>Actual</th>
<th>To Date</th>
<th>To Date %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate/Compile</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Postmortem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Defects Injected

<table>
<thead>
<tr>
<th></th>
<th>Actual</th>
<th>To Date</th>
<th>To Date %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate/Compile</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Development</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Defects Removed

<table>
<thead>
<tr>
<th></th>
<th>Actual</th>
<th>To Date</th>
<th>To Date %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculate/Compile</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Development</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>After Development</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### PSSP & PSP Process Time Recording Log

<table>
<thead>
<tr>
<th>Date</th>
<th>Start Time</th>
<th>Stop Time</th>
<th>Interruption Time</th>
<th>Delta Time</th>
<th>Phase</th>
<th>Comments</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Start Time</th>
<th>Stop Time</th>
<th>Interruption Time</th>
<th>Delta Time</th>
<th>Phase</th>
<th>Comments</th>
</tr>
</thead>
</table>

### PSSP & PSP Defect Recording Log

<table>
<thead>
<tr>
<th>Defect Types:</th>
<th>30 Build, Package</th>
<th>60 Checking</th>
<th>90 System</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 Documentation</td>
<td>40 Assignment</td>
<td>70 Data</td>
<td>100 Environment</td>
</tr>
<tr>
<td>20 Syntax</td>
<td>50 Interface</td>
<td>80 Function</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Student</th>
<th>Date</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Instructor</th>
<th>Spreadsheet #</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Number</th>
<th>Type</th>
<th>Inject</th>
<th>Remove</th>
<th>Fix Time</th>
<th>Fix Defect</th>
<th>Description:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Number</th>
<th>Type</th>
<th>Inject</th>
<th>Remove</th>
<th>Fix Time</th>
<th>Fix Defect</th>
<th>Description:</th>
</tr>
</thead>
</table>

© 1999 A. Winsor Brown BES/MSEE

v1.0 - 06/30/99
First Exercise — Spreadsheet #1

Linear Regression

1. Spreadsheet 1S Prerequisites and References:
   DSE Appendix A2, A7

2. Spreadsheet 1S Requirements
   Develop a spreadsheet to calculate the linear regression size-
estimating parameters for a set of $n$ programs where historical object
LOC and new and changed LOC data are available.
   Linear regression and the required formulas are explained in Section
A7 (but you may use built in hfunctions if you wish).
   Do a graphic frequency distribution of the 1S data. Indicate on it the
mean and +/- 1, 2 and 3 sigma values.
   Do a graphic display of the 1S data sets and super-impose on the
data the linear regression curve.

DSE Appendix A7 abreviated version

Table A28 Linear Regression Example

<table>
<thead>
<tr>
<th>Prg Num</th>
<th>LOC: Xi</th>
<th>LOC: Yi</th>
<th>Xi*Yi</th>
<th>Xi**2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>130</td>
<td>186</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>650</td>
<td>699</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>99</td>
<td>132</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>150</td>
<td>272</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>128</td>
<td>291</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>302</td>
<td>331</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>95</td>
<td>199</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>945</td>
<td>1890</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>368</td>
<td>788</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>961</td>
<td>1601</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sum Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta0</td>
</tr>
<tr>
<td>Beta1</td>
</tr>
</tbody>
</table>
First Exercise — Spreadsheet #1 (cont.)

**Linear Regression**

**Hints:**

- **1S**
  - Use SLOPE(Ys,Xs) for Beta1
  - Use INTERCEPT (Ys,Xs) for Beta0

- **1S Charts**
  - Use 0, ½ maximum of the original independent variable rounded up to the next 100, and the rounded up maximum as the three independent values for the lines;
  - Use the formula with the appropriate values of the betas to calculate the dependent values of the straight lines.

---

**Assignment 1 Details**

**"Fagan's" Exit Criteria**

**Items in product (& standard for acceptance for each)**

**Product requirements**

**Example: My Presentations**

1. **Copyright** (SfA: footer of PowerPoint or Word template)
2. **Version information** (SfA: Version # & Date, “File name”-Page#)
3. **Title** (SfA: one or two lines; 36 pt; centered)
4. **Amount of material** (SfA: max 18 lines of 24 pt text with no graphics):
   - 2-5 Major topics (one line) (No bullet)
   - 2-5 Bullets on 3-4 major bullets with ●
   - 2-5 Sub-bullets on 3-4 sub-bullets with dash lead-in
5. **One or two graphics** (SfA: replacing text lines or not interfering)
6. ...
7. ...
0. The following five items (per paragraphs one and three of C1.5)
   - Complete process data
   - Accurate and self-consistent data
   - Process Report in proper order and format
   - Neat and legible; need not be typed
   - NO cover sheets, binders, or written reports other than those requested

1. PSP0 Project Plan Summary (like C1.4).
2. Time Recording Log (per Table C17, page 658).
3. Defect Recording Log (per Tables C19 and C20, pages 660-661).

4. Source Program Listing

5. Any other requested material:
   Project Plan, if written
   Design Notes, if written
   Test Report
Discuss the CSCL665 Projects

- A diversion to a different set of notes.

- CS665-like SE Projects: System Testing of COCOMOII
  - Discuss project approach
  - Demonstrate COCOMOII
    http://sunset.usc.edu/COCOMOII/cocomo.html#downloads

Personal SE Project Management Process -- Lecture/Week 1

Start Part 2 of PPMP Portion of CS599 Week 1

Discussion

PPMP

- 
  - 

PSSP Exercise 1S Demo and/or Assignment

- 
  - 

Reading

- 
  -
Part 2 of PPMP Portion of CS599 Week 1

Discuss PPMP Exercise, Demonstration & Reading

Complete Overview of PPMP & PSP

Re-Review the Spreadsheet Assignment

CS599 Project and Assignments [Separate Material]

Presentation Outline Detail

Part 4: Discuss PPMP Exercise, Demonstration & Reading

BREAK

Part 5: More on PPMP and PSP
- Describe rest of PPMP's Methods, Techniques & Technologies
- PPMP + PSP
- Personal Quality Management – PSP2s
- Cyclic Personal Process – PSP3
- Personal Process Definition & Development
- Re-Review the Spreadsheet Assignment

BREAK

Part 6: CS599 Project and Assignments [Separate Material]
- System Testing a la IEEE STD J-016
- Using UML as source models for testing
PPMP & PSP Phases and Methods

PPMP0
- Current process
- Time recording
- Defect recording
- Defect type standard

PPMP0.1
- Coding standard
- Size measurement
- Process improvement proposal (PIP)

PSP0
- Code reviews
- Design reviews
- Design templates

PPMP1
- Size estimating
- Test report

PPMP1.1
- Task planning
- Schedule planning

PSP1
- Size estimating
- Test report

PSP1.1
- Task planning
- Schedule planning

PSP2
- Code reviews
- Design reviews

PSP2.1
- Design templates

PSP3
- Cyclic development

PSP/PPMP Steps – Level 0 & 1

Personal Planning Process

PSP0
- Current process
- Time recording
- Defect recording
- Defect type standard

PSP0.1
- Coding standard
- Size measurement
- Process improvement proposal (PIP)

Personal Process Baseline

PSP1
- Size estimating
- Test report

PSP1.1
- Task planning
- Schedule planning
PSP Course Methods, Techniques and Technologies

Level 1 – Personal Planning Process:
Make size, resource, and schedule plans

PSP1
Size estimating
Test report

PSP1.1
Task planning
Schedule planning

Prediction Intervals
Time and schedule estimating
Earned Value Tracking

Linear Regression
PROBE

Level 0 – Personal Process Baseline:
Establish a measured performance baseline

PSP0
Current process
Time recording
Defect recording
Defect type standard

PSP0.1
Coding standard
Size measurement
Process improvement proposal (PIP)

LOC Standard
Object and LOC Counters
Defect Analysis

Linked Lists
Statistics overview and standard deviation
PSP Methods, Techniques & Technologies

Level 3 – Cyclic Personal Process:
Scale up PSP methods to larger projects

PSP3
Cyclic development

Design Verification: Object state machines, Verifying program correctness
Defining and evolving processes

Level 2 – Personal Quality Management:
Practice defect and yield management

PSSP1s Framework – Add Personal Planning

Customer
Need

Define the requirements

Produce the conceptual design.

Estimate the product size. (Chapter 5)

Estimate the resources. (Chapter 6)

Produce the schedule. (Chapter 6)

Delivered Product

Develop the product

Size, resource, schedule data

Analyze the process.

Tracking Reports

Management

Historical size database

Historical productivity database

Resources available

Postmortem

Items

Tasks

Planning

PPMP & PSP 1.1 PROBE & Planning Framework

© 1999 A. Winsor Brown BES/MSEE
PSSP1s Processes – Personal Planning

Earned Value (EV) tracking and planning
- Task and schedule planning, and earned value tracking
- Earned Value spreadsheet exercise

Size estimating
- Approaches
- PROxy Based Estimating (PROBE):
  - Process details
  - Exercise
- Historical databases for PROBE

Resource estimating
- Approaches
- PROBE
- Multiple regression

PSP Steps – Level 2 & 3

Cyclic Personal Process

PSP3
Cyclic development

Personal Quality Management

PSP2
Code reviews
Design reviews

PSP2.1
Design templates
Personal Quality Management – PSP2s

- Code and design reviews
- Design templates
- Design notations
- Design verification approaches
- Focus on yield management:
  - Defect removal strategies
  - Defect prevention strategies
- Quantitative quality management
PSP3 – Cyclic Development

Specifications

Requirements and planning
Produce a conceptual design for the overall system; Estimate its size; Plan the development work.

High-level design
Identify the product's natural divisions; Devise a cyclic strategy.

High-level design review

Cyclic development

Postmortem
Complete Cycle Summary form: with actual time, defect, and size data; Complete Project Plan Summary.

Integration
System test; Use

Product

PSP3 Process Flow

Specify cycle
Establish the specifications for the current cycle

Detailed design and design reviews

[Unit] Test development and reviews
Complete the design for the module test materials and facilities

Implementation and code reviews

Compile

Test

Reassess and recycle;
Record data on the development cycle; Reassess the status against plan and decide to continue as planned or make changes & alert project management

“essentially a PSP2.1 process”

System test;
Use

Product

PSP3 – Cyclic Development

Specifications

Requirements and planning
Produce a conceptual design for the overall system; Estimate its size; Plan the development work.

High-level design
Identify the product's natural divisions; Devise a cyclic strategy.

High-level design review

Cyclic development

Postmortem
Complete Cycle Summary form: with actual time, defect, and size data; Complete Project Plan Summary.

Integration
System test; Use

Product

PSP3 Process Flow

Specify cycle
Establish the specifications for the current cycle

Detailed design and design reviews

[Unit] Test development and reviews
Complete the design for the module test materials and facilities

Implementation and code reviews

Compile

Test

Reassess and recycle;
Record data on the development cycle; Reassess the status against plan and decide to continue as planned or make changes & alert project management

“essentially a PSP2.1 process”

System test;
Use

Product
PSP3 – Cyclic Development

Process scalability
Development strategies to support cycle selection
PSP Processes

PPMP/PSP0
- Coding standard
- Size measurement
- Process improvement proposal (PIP)

PPMP/PSP1
- Size estimating
- Test report

PPMP/PSP1.1
- Task planning
- Schedule planning

PPMP/PSP2
- Code reviews
- Design reviews

PPMP/PSP2.1
- Design templates
- Checklists for reviews
- Defining and evolving processes

PSP3
- Cyclic development
- Design verification: Object state machines, verifying program correctness

Personal Planning Process

Personal Process Baseline

Cyclic Personal Process

Personal Quality Management

PROBE
- LOC DB

LOC Standard
- Object and LOC Counters
- Defect Analysis
- Prediction Intervals
- Correlation

Time and schedule estimating
- Earned Value Tracking
- Goal-Question-Metric Paradigm
- Multiple Regression

Process benchmarking
- Yield management
- Defect removal and prevention strategies
- Design Notations
- Operational scenarios
- Functional, State & Logic specifications

Defining and evolving processes

Design Verification: Object state machines, verifying program correctness

Defining and evolving processes
Process Definition and Development

Process development strategy

Define, test and adjust

Definition levels

Defining process steps

Process definition guidelines

Process evolution

Process development process

Process Definition and Development (cont.)

Process Definition Guidelines

Start with your current process.

- make minimal initial enhancements
- only define things you know reasonably well how to do
- if you know a step well, focus elsewhere
- if a step is confusing, experiment or talk to someone who knows how to do it
- expect to get it wrong the first time
- plan to prototype and test
- plan for perpetual enhancement
- make improvements in small steps
Process Evolution

To evolve and improve your process
- it must be defined
- it must reasonably represent what you do

You must
- know where you want to go
- be willing to experiment
- observe and measure your own performance

Expect process evolution to take time.

The 4 Processes

- what you do
- what you want to do
- what you are supposed to do
- what you think you do
Start By Converging

- what you do
- what you think you do
- your initial process
- what you are supposed to do
- what you want to do

Process Development Process

Start with a simplified process definition

Plan and measure your work

- record development time per process category
- count the numbers of product items by category
- define productivity measures
- plan every process development
- keep a record of each process development
- produce a summary report for each process development
A Process Development Process

1. Process development is an evolutionary process.

2. You must understand your process before you can properly define it.

3. Process development is a process that should also be defined, measured, and improved.
At Course Conclusion

You will have practiced the key elements of a level 5 industrial process.

You will understand which methods are most effective for you.

You will do better work.

You will have long-term improvement goals.

Messages to Remember from WH's Lecture 1

1 - The PSP is a defined process to help you do better work.

2 - Once you have completed this course, you will know how to adjust and extend the PSP to meet your future needs.

3 - In using PSP0, your principal objective is to gather and report accurate and complete data on your work.
Exercise Spreadsheet 1S Review

1. **Spreadsheet 1S Prerequisites and References:**
   DSE Appendix A2, A7

2. **Spreadsheet 1S Requirements**
   Develop a spreadsheet to calculate the linear regression size-estimating parameters for a set of \( n \) programs where historical object LOC and new and changed LOC data are available.

   Linear regression and the required formulas are explained in Section A7 (but you may use built in hfunctions if you wish).

   Do a graphic frequency distribution of the 1S.A data. Indicate on it the mean and +/- 1, 2 and 3 sigma values.

   Do a graphic display of the 1S data sets and super-impose on the data the linear regression curve.
Standard Deviation & Linear Regression

DSE Appendix A7 abbreviated version

\[
\beta_1 = \frac{\sum_{i=1}^{n} x_i y_i - nx_{avg} y_{avg}}{\sum_{i=1}^{n} x_i^2 - n(x_{avg})^2}
\]

\[
\beta_0 = y_{avg} - \beta_1 x_{avg}
\]

<table>
<thead>
<tr>
<th>Prg Num</th>
<th>LOC: Xi</th>
<th>LOC: Yi</th>
<th>Xi*Yi</th>
<th>Xi**2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>130</td>
<td>186</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>650</td>
<td>699</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>99</td>
<td>132</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>150</td>
<td>272</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>128</td>
<td>291</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>302</td>
<td>331</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>95</td>
<td>199</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>945</td>
<td>1890</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>368</td>
<td>788</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>961</td>
<td>1601</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table A28 Linear Regression Example

2. Spreadsheet 1S Requirements (cont.)

Do a graphic frequency distribution of the 1S.A data.

Indicate on it the mean and +/- 1, 2 and 3 sigma values.

Do a graphic display of the 1S.B data sets and superimpose on the data the linear regression curve.
## Linear Regression

**Hints:**

- **1S calculations**
  - Use `SLOPE(Ys,Xs)` for Beta1
  - Use `INTERCEPT (Ys,Xs)` for Beta0

- **1S charts**
  - Use 0, \( \frac{1}{2} \) maximum of the original independent variable rounded up to the next 100, and the rounded up maximum as the three independent values for the lines;
  - Use the formula with the appropriate values of the betas to calculate the dependent values of the straight lines.