

## WVU LDCSEE CS 430

### Project Scheduling and Tracking

copyright © 1996, 2001, 2005  
R.S. Pressman & Associates, Inc.

#### For University Use Only

May be reproduced ONLY for student use at the university level  
when used in conjunction with *Software Engineering: A Practitioner's Approach*.  
Any other reproduction or use is expressly prohibited.

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

1

---

---

---

---

---

---

---

---

---

---

### Why Are Projects Late?

- an unrealistic deadline established by someone outside the software development group
- changing customer requirements that are not reflected in schedule changes;
- an honest underestimate of the amount of effort and/or the number of resources that will be required to do the job;
- predictable and/or unpredictable risks that were not considered when the project commenced;
- technical difficulties that could not have been foreseen in advance;
- human difficulties that could not have been foreseen in advance;
- miscommunication among project staff that results in delays;
- a failure by project management to recognize that the project is falling behind schedule and a lack of action to correct the problem

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

2

---

---

---

---

---

---

---

---

---

---

### Scheduling Principles

- compartmentalization—define distinct tasks
- interdependency—indicate task interrelationship
- effort validation—be sure resources are available
- defined responsibilities—people must be assigned
- defined outcomes—each task must have an output
- defined milestones—review for quality

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

3

---

---

---

---

---

---

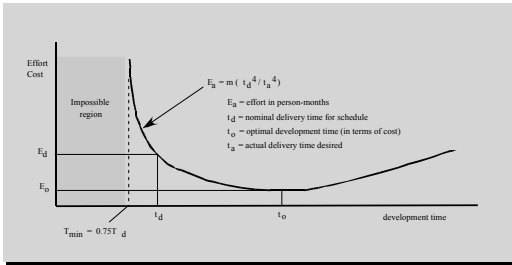
---

---

---

---

## Effort and Delivery Time



These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

4

---

---

---

---

---

---

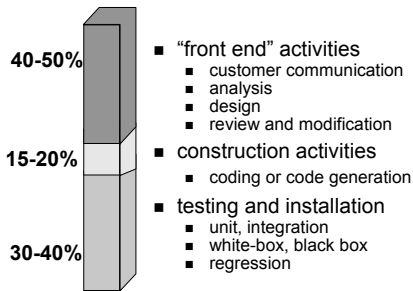
---

---

---

---

## Effort Allocation



These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

5

---

---

---

---

---

---

---

---

---

---

## Defining Task Sets

- determine type of project
- assess the degree of rigor required
- identify adaptation criteria
- select appropriate software engineering tasks

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

6

---

---

---

---

---

---

---

---

---

---





## Progress on an OO Project-II

- **Technical milestone: OO programming completed**
  - Each new class has been implemented in code from the design model.
  - Extracted classes (from a reuse library) have been implemented.
  - Prototype or increment has been built.
- **Technical milestone: OO testing**
  - The correctness and completeness of OO analysis and design models has been reviewed.
  - A class-responsibility-collaboration network (Chapter 8) has been developed and reviewed.
  - Test cases are designed and class-level tests (Chapter 14) have been conducted for each class.
  - Test cases are designed and cluster testing (Chapter 14) is completed and the classes are integrated.
  - System level tests have been completed.

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

13

---

---

---

---

---

---

---

---

---

---

## Earned Value Analysis (EVA)

- **Earned value**
  - is a measure of progress
  - enables us to assess the “percent of completeness” of a project using quantitative analysis rather than rely on a gut feeling
  - “provides accurate and reliable readings of performance from as early as 15 percent into the project.” [FLE98]

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

14

---

---

---

---

---

---

---

---

---

---

## Computing Earned Value-I

- The *budgeted cost of work scheduled* (BCWS) is determined for each work task represented in the schedule.
  - $BCWS_i$  is the effort planned for work task  $i$ .
  - To determine progress at a given point along the project schedule, the value of BCWS is the sum of the  $BCWS_i$  values for all work tasks that should have been completed by that point in time on the project schedule.
- The BCWS values for all work tasks are summed to derive the *budget at completion*, BAC. Hence,
  - $BAC = \sum (BCWS_k)$  for all tasks  $k$

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner's Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005

15

---

---

---

---

---

---

---

---

---

---

## Computing Earned Value-II

- Next, the value for *budgeted cost of work performed* (BCWP) is computed.
  - The value for BCWP is the sum of the BCWS values for all work tasks that have actually been completed by a point in time on the project schedule.
- “the distinction between the BCWS and the BCWP is that the former represents the budget of the activities that were planned to be completed and the latter represents the budget of the activities that actually were completed.” [WIL99]
- Given values for BCWS, BAC, and BCWP, important progress indicators can be computed:
  - Schedule performance index,  $SPI = BCWP/BCWS$
  - Schedule variance,  $SV = BCWP - BCWS$
  - SPI is an indication of the efficiency with which the project is utilizing scheduled resources.

---

---

---

---

---

---

---

---

---

---

## Computing Earned Value-III

- Percent scheduled for completion =  $BCWS/BAC$ 
  - provides an indication of the percentage of work that should have been completed by time  $t$ .
- Percent complete =  $BCWP/BAC$ 
  - provides a quantitative indication of the percent of completeness of the project at a given point in time,  $t$ .
- *Actual cost of work performed*, ACWP, is the sum of the effort actually expended on work tasks that have been completed by a point in time on the project schedule. It is then possible to compute
  - Cost performance index,  $CPI = BCWP/ACWP$
  - Cost variance,  $CV = BCWP - ACWP$

---

---

---

---

---

---

---

---

---

---