Software Project Management

Lecture # 8

What are we studying today?

- Chapter 24 - Project Scheduling
  - Project Scheduling
  - Timeline charts
  - Schedule tracking
  - Earned Value Analysis

Scheduling

- Two Project Scheduling methods that can be applied to software development:
  - Program evaluation and review technique (PERT)
  - Critical Path Method (CPM)
- Both are driven by Info already developed:
  - Estimates of effort
  - A decomposition of product function
  - The selection of appropriate process model and task set
  - Decomposition of tasks

Timeline Charts

- When creating software project schedule, the planner begins with a set of tasks (work breakdown structure)
- If automated tools are used, the work breakdown is input as a task network
- Effort, duration and start date are then input for each task
- As a result of this input, a timeline chart or Gantt chart is generated

Tracking Schedule

- Accomplished in a no. of ways
  - Conducting periodic project status meetings in which each team member reports progress & problems
  - Evaluating results of all reviews conducted throughout the s/w engg process
  - Determining whether formal project milestones have been accomplished by scheduled date
  - Comparing actual start date to planned date for each project task
  - Meeting practitioners to get their assessment of progress to date and problems encountered
  - Use earned value analysis to assess progress quantitatively

Earned Value Analysis

- Earned Value Analysis (EVA) provides a quantitative indication of progress of a project.
- Humphrey discusses earned value as:
  - "*The earned value system provides a common value scale for every [software project] task, regardless of the type of work being performed. The total hours to do the whole project are estimated and every task is given an earned value based on its estimated percentage of the total*"

Earned Value Analysis (Contd.)

- It compares the PLANNED work with the actually COMPLETED work, to determine if *COST* , *SCHEDULE, and WORK ACCOMPLISHED* are progressing as planned.

- EVA is accomplished using Earned Value Management.

Earned Value Management (EVM)

- EVM is a project management technique for measuring project progress.
- If applied properly, it provides an early warning of performance and
  - Improves definition of scope
  - Communicates progress to stakeholders
  - and keeps the project team focused on achieving progress.

Earned Value Management (EVM)

- EVM implementation includes
  - A project plan that identifies work to be accomplished or WBS,
  - A valuation of planned work called Planned Value (**PV**) or Budgeted Cost of Work Scheduled (**BCWS**)
  - Predefined “earning rules” (a.k.a. metrics) to quantify the accomplishment of work called Earned Value (**EV**) or Budgeted Cost of Work Performed (**BCWP**)
    - BCWP value is sum of the BCWS values for all work tasks that have actually been completed by a point in time on the project schedule

Earned Value Management

- '*Wilkens*' gives distinction between BCWS and BCWP as:
  - *BCWS represents the budget of activities that were planned to be completed*
  - *and BCWP represents the budget of activities that were actually completed*
- EVM for large projects includes many more features such as indicators and forecast of cost performance and schedule performance.
- Important progress indicators can be computed based on BCWS, BAC, BCWP values

Earned Value Management

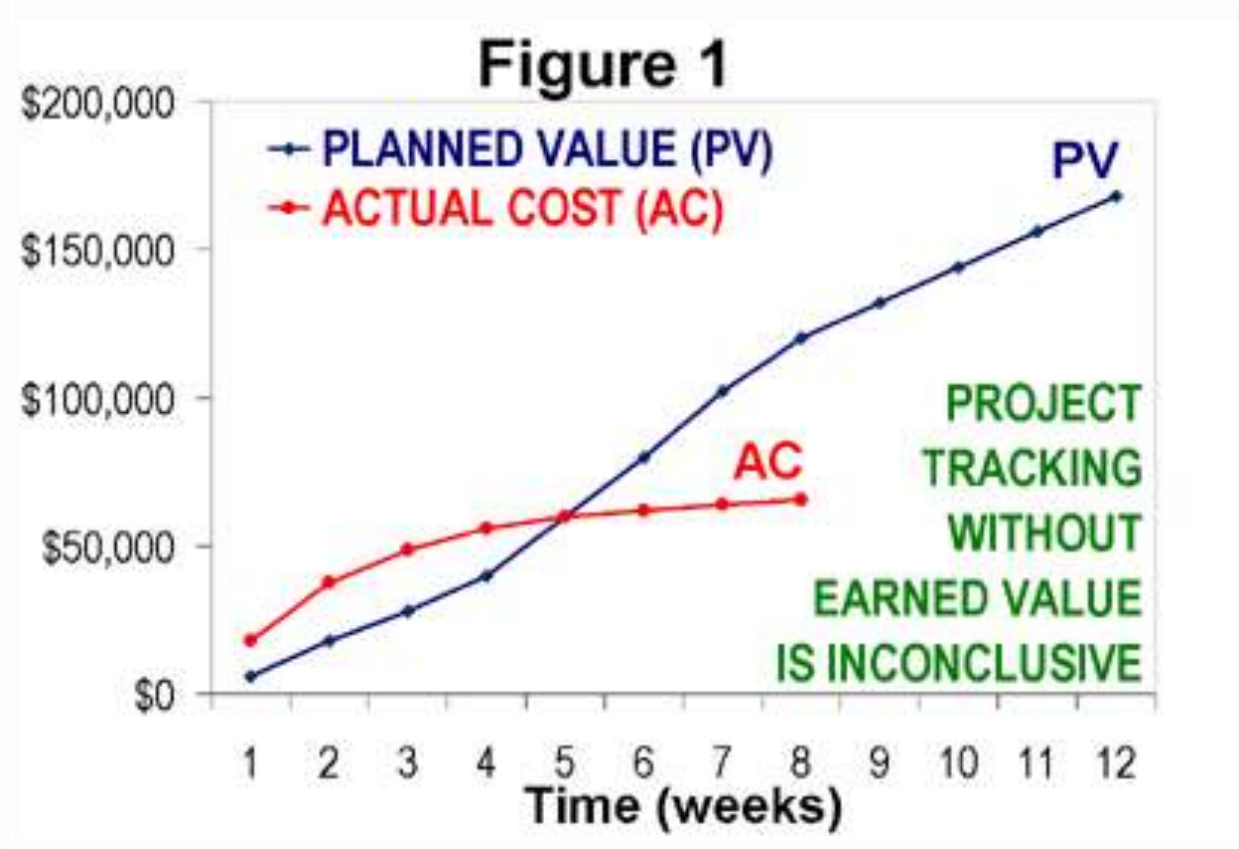
- SPI = BCWP/BCWS
  - Schedule performance index indicates efficiency with which project is using scheduled resources. SPI value 1.0 indicates efficient execution of project schedule
- SV = BCWP – BCWS
  - Schedule variance is simply an absolute indication of variance from planned schedule
- Percent scheduled for completion= BCWS/BAC
  - Where BAC is Budget at completion computed as: BAC = $\sum(BCWS_k)$ for all k tasks

Earned Value Management

- Percent complete= BCWP/BAC
  - Provides indication of percent of completeness of the project at a given point in time t.
- CPI = BCWP/ACWP
  - Cost performance index value close to 1.0 indicates project is within its defined budget
  - Actual Cost of Work Performed ACWP is sum of effort actually expended on work tasks that have been completed by a point in time on project schedule
- CV = BCWP – ACWP
  - Cost Variance is an absolute indication of cost savings against planed or shortfall at a particular stage of project

Project Tracking without EVM

Blue line= the cumulative budget for this project as a function of time.
Red line = the cumulative actual cost of the project through week 8.



Project Tracking without EVM

- To those unfamiliar with EVM, it might appear that this project was over budget through week 4 and then under budget from week 6 through week 8.
- However, this chart does not indicate how much work has been accomplished during the project.
- If the project was actually completed at week 8, then the project would actually be well under budget and well ahead of schedule.
- If, on the other hand, the project is only 10% complete at week 8, the project is significantly over budget and behind schedule.
- A method is needed to measure technical performance (progress) objectively and quantitatively, and that is what EVM accomplishes.

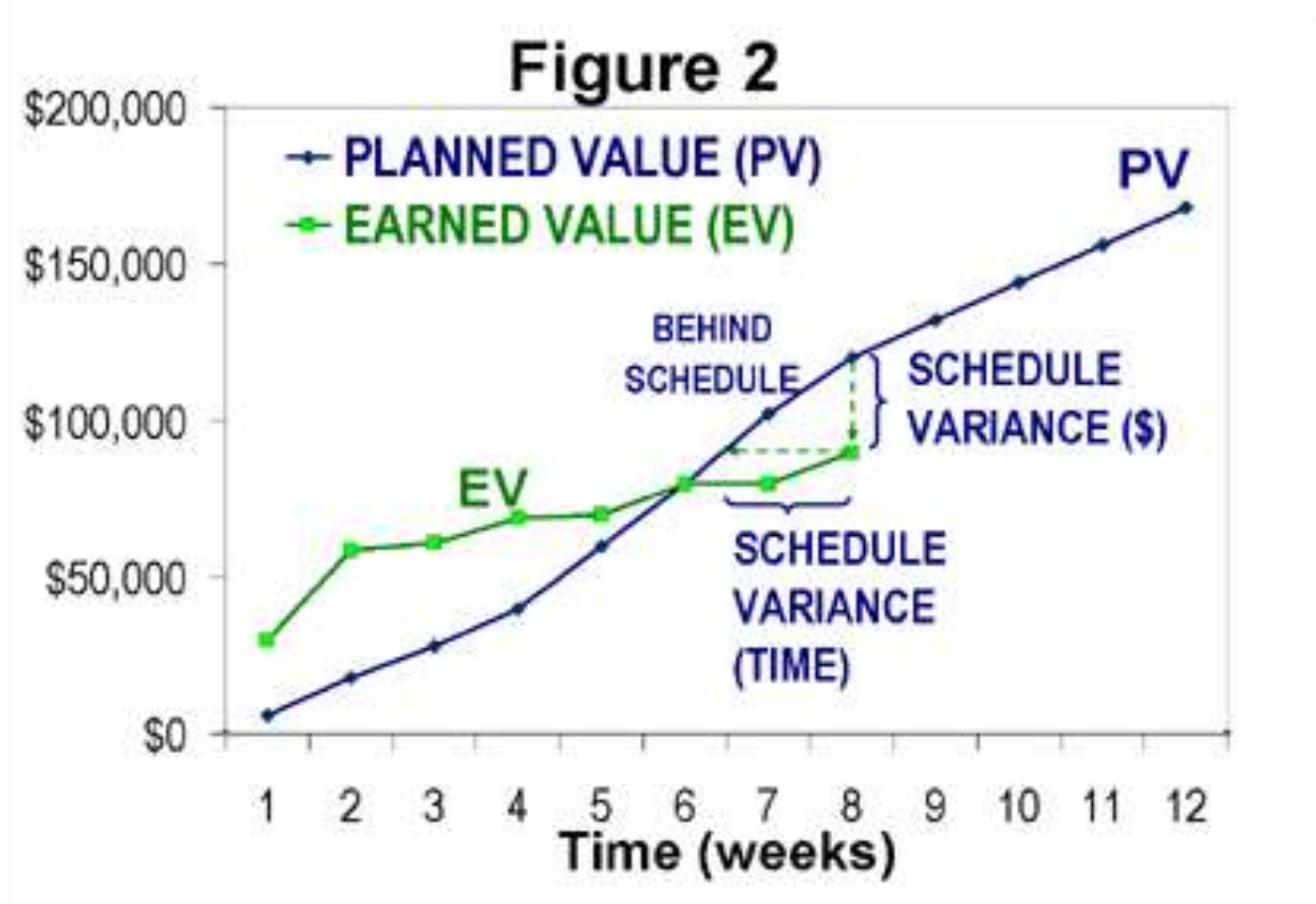
Project Tracking with EVM

- Consider the same project, except this time the project plan includes pre-defined methods of quantifying the accomplishment of work.
- At the end of each week, the project manager identifies every detailed element of work that has been completed, and sums the PV for each of these completed elements.
- Earned value may be accumulated monthly, weekly, or as progress is made.

$$\mathrm{EV} = \sum_{\mathrm{Start}}^{\mathrm{Current}} \mathrm{PV(Completed)}$$

Project Tracking with EVM

Green line = EV curve, Blue line = PV curve



Project Tracking with EVM

- The chart indicates that progress started more rapidly than planned, but slowed significantly and fell behind schedule at week 7 and 8.
- This chart illustrates the schedule performance aspect of EVM.

References

- Today’s lecture contents were taken from
  - Chapter 24 – “Software Engineering, A Practitioner's Approach” by Roger Pressman
  - Wikipedia