The Need for Planning

- Work more effectively
  - do things in a productive order
  - don’t miss important steps
- Meet commitments
  - know what you must do
  - know when it will be done
- Better work
  - don’t skip steps due to schedule pressure
Balanced Plan

- Unbalanced workload leads to scheduling problems.
- Overcommitted engineers can delay the entire team.
- Results in shortest possible schedule

Tracking Progress

- Tasks don’t necessarily get completed in the order originally planned.
- Need to know where you are.
- Need to know how you got there.
- Intuitive assessments of progress are rarely correct.
  - People tend to believe that they are 90% complete when they are at best 50% complete.
Planning in Detail

• Assume that you estimate 250 hours for a job.
• Assume that this estimate is accurate.
• After working 150 hours, where are you?
• If you plan the job in 10-hour increments, the maximum uncertainty is 10 hours.

Unplanned Tasks

• Normally, unplanned tasks earn no Earned Value.
• Unless the job is similar to one done before, you will probably run into tasks you did not anticipate.
• With experience, these will be small. If not, then re-planning is required.
Unplanned Tasks (2)

- Until you have sufficient experience, and have a well-defined process, reserve 5 to 10 percent for unplanned tasks.
- Place 2-hour “management and miscellaneous” (M&M) tasks throughout your schedule.
- When you perform an unplanned task, take credit for one of the M&M’s.

The Planning Process
Planning Process (2)

1. Develop Conceptual Design to identify parts.
2. Estimate the sizes of the parts.
3. Enter the sizes of the parts for the current cycle on the SUMS form. Add additional products such as test plans and requirements documents.
4. List the tasks required for each product and estimate the time. Where possible assign times to individual engineers. Use TSPi tool to generate TASK and SCHEDULE forms.
5. Quality plan deleted.

Planning Process (3)

6. Each engineer make’s their own individual TASK and SCHEDULE forms.
7. Use TSPi tool to generate individual TASK and SCHEDULE form with PV and completion times.
8. Balance the workload. Revise individual TASK and SCHEDULE forms as necessary.
9. Use TSPi tool to roll up individual plans into overall plan.
Detailed Planning Process

Planning Script
Planning Script (2)

<table>
<thead>
<tr>
<th>Purpose</th>
<th>To guide a team through producing individual and team task and schedule plans for development cycle 1</th>
</tr>
</thead>
</table>
| Entry Criteria | • The team has a development strategy and conceptual design  
• The students have read textbook Chapter 5 |
| General | The task plan defines the  
• Time required to perform each process task  
• Rough order in which the tasks will be performed  
• Planned value of each task  
The schedule plan gives  
• Each engineer’s planned time for each project week  
• The total planned team hours by week  
• The anticipated completion week for each task  
• The planned value for each task  
If the task and schedule plans indicate the project will not be completed on time, readjust the strategy and replan. |

Planning Script (3)

<table>
<thead>
<tr>
<th>Step</th>
<th>Activities</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Planning Overview | The instructor describes the planning process  
• The task and schedule plans and how they are produced. |
| 2    | Enter the Size Estimates in form STRAT | Starting with the conceptual design and STRAT form produced in the strategy phase, the planning manager leads the team in  
• Identifying any other products to be produced and their size  
• Recording the STRAT form and other size data in SUMS |
| 3    | Produce the Task Plan | The planning manager leads the team through  
• Producing a task list with team and engineer time estimates  
• Entering these data in the TASK form |
| 4    | Produce the Schedule Plan | The planning manager obtains the estimated number of hours each team member plans to spend on the project each week and  
• Enters the weekly hours in the SCHEDULE form  
• Produces the team TASK and SCHEDULE forms  
• Reworks the plan if the hours are inadequate |
Planning Script (4)

<table>
<thead>
<tr>
<th></th>
<th>Produce the Quality Plan</th>
<th>Quality plan deleted.</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Produce the Individual Engineer Plans</td>
<td>The planning manager helps the engineers make personal plans.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Allocating the tasks among the team members</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Estimating the time to perform each task</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Entering the data in the TASK and SCHEDULE forms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Producing the planned-value schedule and task completion dates</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Balance Team Workload</th>
<th>The planning manager leads the team through</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>• Identifying workload imbalances</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Reallocating tasks to minimize the schedule</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Producing balanced engineer plans</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Producing the consolidation team plan</td>
</tr>
</tbody>
</table>

Exit Criteria

- Completed team and engineer TASK and SCHEDULE forms
- Completed SUMP forms
- Updated project notebook

SUMS Instructions

**Purpose**

- This form summarizes the data for product size
- At the lowest level, it summarizes all of the part size data
- At higher levels, it summarizes the size data for an assembly and its parts.

**General**

- Use this form to hold size data for the parts of an assembly.
- If you are using the TSPi tool, these data are automatically rolled up to the next-level SUMP form.
- If you are not using the TSPi tool, enter the totals from this form in the appropriate places in the assembly-level SUMP form.
- Note: the number of rows in this form is variable, depending on the number of products and parts.

**Header**

- Enter your name, date, team name, and instructor's name.
- Name the part of assembly and its level.
- Enter the cycle number.
SUMS Instructions (2)

<table>
<thead>
<tr>
<th>Plan/Assembly/Actual</th>
<th>Check whether this form is for plan, assembly, or actual</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Plan: In system-level planning, use SUMS to hold the</td>
</tr>
<tr>
<td></td>
<td>estimated sizes of the system and all its parts.</td>
</tr>
<tr>
<td></td>
<td>• Assembly: Use a separate SUMS form for the estimated</td>
</tr>
<tr>
<td></td>
<td>size data for the parts in each assembly.</td>
</tr>
<tr>
<td></td>
<td>• Actual: Use a SUMS form for the actual size data for</td>
</tr>
<tr>
<td></td>
<td>the parts of each assembly.</td>
</tr>
</tbody>
</table>

| Product and Part Names | • List the name or other identification of each product or part. |
|                       | • It is also a good idea to number the product components, particularly if there will be very many of them. |
|                       | • Include system-level products such as SRS and HLD. |

<table>
<thead>
<tr>
<th>Size Measure</th>
<th>• Enter the size measures used for each product item listed.</th>
</tr>
</thead>
</table>

| Program Size | • For each product, enter the actual or estimated program base, deleted, added, modified, reused, and new and changed LOC. |

| Totals | • Enter the totals for each size category and column. |

Parts and Assemblies

- The TSPi uses the generalized terms Parts and Assemblies.
- For our project we will use the terms Class as an Assembly and Member Function as a Part.
- Consider the Class declaration (the .h file) also as a Part.
Tasks

• The TSPi uses an elaborate life-cycle consisting of:
  – Requirements
  – High Level Design
  – Detailed Design
  – Code, Compile, Unit Test (each its own task)
  – Build and Integration
  – System test

Tasks (2)

• For this project you are free to adapt this life-cycle as you see fit.
• Minimum artifacts (Use Rational Rose)
  – Requirements documented as Use Cases.
  – Sequence Diagrams as needed
  – Class diagram
    • All attributes identified
    • All methods identified
      – Arguments
      – Return value
      – Contract
Reviews and Inspections

- Reviews and Inspections have a high payoff in terms of both the final quality of the product and the productivity of the team.
- Errors found earlier in the life-cycle are easier to fix than those found later.
- Reviews and inspections will be discussed in later lectures.

Making the Schedule

- Lay the tasks out in the order in which they must be accomplished. (I.e. design comes before code, code comes before test, etc.)
- Compute the cumulative time required. (The TSPi tool does this.)
- Compare with cumulative planned time to be worked each week to get completion times for each task.
Balancing The Schedule

• Shift work from those engineers who are finishing later to those engineers who are finishing earlier.

What if Schedule is Too Long.

• DO NOT ADJUST ESTIMATES TO MAKE IT FIT!!!!!
• Go back to your high-level design and reduce functionality.
Tracking Progress

• When a task is complete, that task’s Planned Value (PV) becomes Earned Value (EV).
• Compare cumulative PV with cumulative EV for each week.
• EV > PV, ahead of schedule
• EV < PV, behind schedule

Tracking Progress (2)

• If you record actual time spent, you can also compute Actual Value (AV).
• EV > AV, you are more efficient than planned.
• EV < AV, you are less efficient than planned.