Estimating With Objects - Part X

Contents

This column is the tenth in a series about estimating. If you have read the prior columns, you now know how object-oriented methods can help you make good estimates and plans. After this there is only one further column which shows some data on how the PROBE method has helped engineers make better plans.

If you are new to this series of columns on Estimating with Objects, the first was in the July 1996 Object Currents issue. The prior columns in this series gave an overview of estimating and defined the steps needed to make size and resource estimates. If you have not read these earlier columns, you should look at them first to understand the context for this discussion and to see how these various estimating topics relate. To repeat what I have said in previous columns, the estimating method described here is called PROBE. If you want to quickly learn more about PROBE, you should read my book A Discipline for Software Engineering, from Addison Wesley. This book introduces the Personal Software Process (PSP)^{SM}, which is an orderly and defined way for software engineers to do their work.

This month's column concludes the discussion of how object-oriented techniques can help you estimate and plan your work. To make a project plan, you need a resource estimate and, to estimate resources, you need to estimate the size of the product you plan to build. Also, to make good estimates, you need historical data on the sizes and development times for the programs you have previously written. The previous columns described how to gather these data and how to use them to make size and resource estimates, judge the accuracy of these estimates, and produce a project schedule. In this column we talk about tracking the work and projecting when you will likely get done.

Earned value tracking

The earned value method was developed as a way to track projects and accurately estimate their completion dates. Earned value is particularly useful when your plan is reasonably complete and accurate, but the order of the tasks changes as you do the work. With earned value, you assign a value to each task, and then use this value to track progress. Whenever you complete a task, you earn its value. This is true whether the task is requirements, planning, design, or test. As long as

the task was planned, and as long as you had estimated how long it would take, you can use earned value to track your progress, regardless of the order in which you do the tasks.

To establish the earned value, start with the project plan and calculate the percent each task is of the total job. That is, if the total job was to take 100 hours, and a task was planned to take 12 hours, that task would get a value of 12. During planning, we call this the planned value. When you do the work, you earn this value when you complete that task. Thus we talk about planned value during planning and earned value as we track the work. Each task earns its earned value regardless of when the task is done and regardless of how long it actually takes. Thus, if you are late with one task and early with another of comparable value, you will still be on schedule. Also, there is no bonus for taking less time and no penalty for taking more to complete a task.

To make an earned value plan, assign a planned value to every task. For every week, note the value for the tasks to be completed that week. Then total up these planned values over the project schedule to get a planned value for every week of the plan.

Next, as you do the work, assign earned value credit for each task that is done that week. This gives you an earned value for every project week. You can then compare this with the planned value. This provides an earned value schedule and tracking system that you can use to continually measure progress. You can see where you stand with respect to the plan, and you can precisely identify your schedule problems.

Only completed tasks get credit

A common difficulty with project tracking is that tasks are often vaguely defined. Examples would be coding 90% done, or design complete. Unless you define tasks precisely, you cannot determine project status. For these two examples, more precise definitions would be: 9 of 10 modules coded and released to test, or the design completed, documented, and inspected.

With earned value, you only earn value for a task when it is completed. There is no partial credit. When the task is done you earn its full value. Thus, even if a task is reported to be 99% complete, you earn nothing until it is actually done.

If you object to this practice of not giving partial credit, there is an easy fix. All you need to do is break these larger tasks into smaller parts. Then you get credit for each smaller task when it is completed. If the 99% complete job had 10 equal parts, you could then take 90% credit when you had completed 9 of the 10 parts. A reasonable rule of thumb on task granularity is to aim for 2 to 4 task completions per week.

An earned value example

To see how this works, lets establish a plan, make the earned value projections for that plan, and then track progress against the plan. Last month, we used an example plan for 3 engineers, on a

project where they had 19 tasks to do over a period of about 20 weeks. The Task Planning Template for last month is shown in Table 1 and the Schedule Planning Template for all 3 engineers is shown in Table 2. If you do not remember how these templates were produced, I suggest you look at last month's column.

In Table 1 gives the sum of these planned values. As you can see, the cumulative value totals 100% when all the work is done.

Table 1. An Example Task Planning Template

Task	**Task Name**	**Task Hours**		**Plan**	**Week**	**Plan**	**Cum.**
No.		Unit	Cum.	Week	Done	Value	PV
1	Requirements	140	140	3		11.80	11.80
2	Requirements Inspection	22	162	3		1.85	13.66
3	Overall Design	170	332	6		14.33	27.99
4	Design C1	16	348	6		1.35	29.34
5	Design C2	12	360	6		1.01	30.35
6	Design C3	22	382	7		1.85	32.21
7	C1 Design Inspection	4	386	7		0.34	32.55
8	C2 Design Inspection	4	390	7		0.34	32.88
9	C3 Design Inspection	6	396	7		0.51	33.39
10	Implement C1	90	486	8		7.59	40.98
11	Implement C2	76	562	9		6.41	47.39
12	Implement C3	110	672	11		9.27	56.66
13	C1 Code Inspection	8	680	11		0.67	57.34
14	C2 Code Inspection	10	690	11		0.84	58.18
15	C3 Code Inspection	16	706	12		1.35	59.53
16	Test C1	36	742	12		3.04	62.56
17	Test C2	66	808	13		5.56	68.13

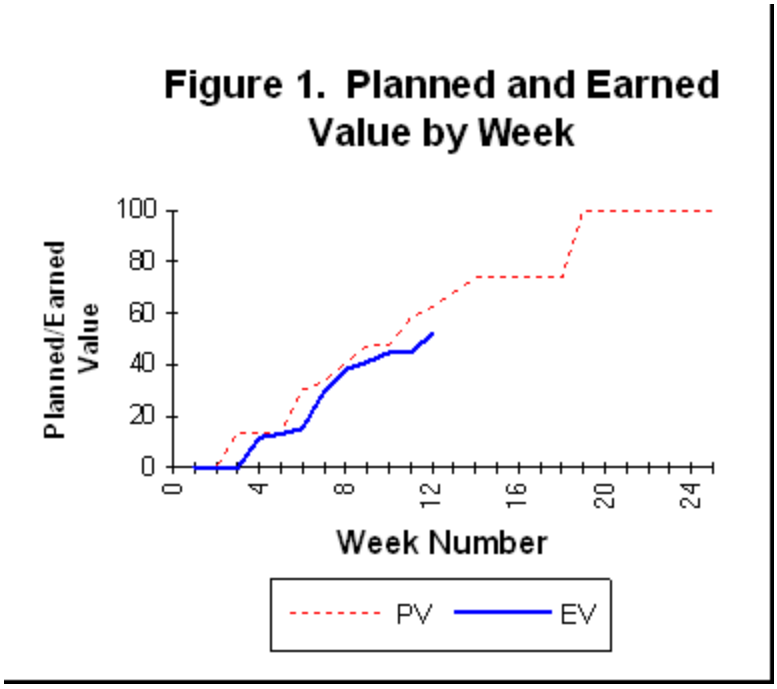
18	Test C3	68	876	14		5.73	73.86
19	System Test	310	1186	19		26.14	100.0
	Total	1186				100.0	

As you do the work, note the week each task is completed on the Task Planning Template. Also, enter on the Schedule Planning Template the earned value for those tasks in the weeks they were completed. By tracking these values, you can plot progress every week, as shown in Figure 1.

The principles of project tracking

While the tracking system described here is simple and works very well, it depends on some assumptions. First, you must start with a reasonably good plan. All known tracking methods track the work against the plan. If the plan is not accurate, there is no way to accurately track progress.

Second, you must track against the original plan. This is a key point. If you keep changing the plan, you cannot track against it. You would then always appear to be on schedule. Thus all plans must be dated and all tracking is done as of a given date. Then you can tell where you are against the original plan.



Third, when the plan no longer represents the way the work is being done, make a new plan. To do this, you must reestimate the product to be built, reestimate the resources required, and

produce a new schedule. While this is not hard to do, it is important to be disciplined in planning. When the project is seriously off schedule, don't just make a minor plan adjustment. Rethink the plan and make sure the new plan is better than the one you had. You now have data on how long the work has taken so far. Use these data to make a better plan.

Table 2. An Example Schedule Planning Template

Week	**Planned Hours**		**Actual**	**Cumulative**	
Number	**Weekly**	**Cumulative**	**Hours**	**Plan Value**	**EV**
1	63	63		0	
2	63	126		0	
3	63	189		13.66	
4	63	252		13.66	
5	63	315		13.66	
6	63	378		30.35	
7	63	441		33.39	
8	63	504		40.98	
9	63	567		47.39	
10	63	630		47.39	
11	63	693		58.18	
12	63	756		62.56	
13	63	819		68.13	
14	63	882		73.86	
15	63	945		73.86	
16	63	1008		73.86	
17	63	1071		73.86	
18	63	1134		73.86	

19	63	1197		100.0	
20	63	1260			
Total	1260				

Tracking projects that change

With modern technology, the work is rarely done precisely as planned. Regardless of how much care you take in planning, you cannot plan precisely what will happen. As you do the work, you will think of new approaches. There will be new technical opportunities, or you might run into unexpected problems. Thus, the tracking method must be adaptable to changes in the way the work will be done.

As long as the job is done pretty much the way it was planned, the earned value method can adapt for changes in the order of the tasks. Some tasks may be done sooner than expected, others may be completed later, and the task times may vary considerably. The earned value for each task, however, is not changed.

Estimating project completion

Table 3. Tracking Progress with the Task Planning Template

Task	Task Name	Task Hours		Plan	Week	Plan	Cum.
No.		Unit	Cum.	Week	Done	Value	PV
1	Requirements	140	140	3	4	11.80	11.80
2	Requirements Inspection	22	162	3	6	1.85	13.66
3	Overall Design	170	332	6	7	14.33	27.99
4	Design C1	16	348	6	5	1.35	29.34
5	Design C2	12	360	6	8	1.01	30.35
6	Design C3	22	382	7	9	1.85	32.21
7	C1 Design Inspection	4	386	7	6	0.34	32.55
8	C2 Design Inspection	4	390	7	9	0.34	32.88

9	C3 Design Inspection	6	396	7	10	0.51	33.39
10	Implement C1	90	486	8	8	7.59	40.98
11	Implement C2	76	562	9	12	6.41	47.39
12	Implement C3	110	672	11		9.27	56.66
13	C1 Code Inspection	8	680	11	9	0.67	57.34
14	C2 Code Inspection	10	690	11	12	0.84	58.18
15	C3 Code Inspection	16	706	12		1.35	59.53
16	Test C1	36	742	12	10	3.04	62.56
17	Test C2	66	808	13		5.56	68.13
18	Test C3	68	876	14		5.73	73.86
19	System Test	310	1186	19		26.14	100.0
	Total	1186				100.0	

The earned-value method is very helpful when the task order changes. When the tasks themselves change, however, the job is substantively different from the plan. Since the plan is no longer valid, there is no way the earned value method can provide good tracking against it. If the new tasks were relatively small, you could still make minor adjustments and use the original earned value plan. If the task changes are significant, however, you have no choice but to make a new plan. You should generally make a new plan as soon as it is clear that the completion date must change or whenever the plan ceases to be a useful guide to your daily work.

With the earned value method, you can estimate when you will likely get done. Assume that the project has so far progressed as shown in Table 3 and Table 4 and also assume that the current rate of earned value progress will continue to the end. That is, you had planned to earn 5.26 earned value per week, but have only earned 4.32 so far. You thus assume that the 4.32 rate will continue to the end of the project. As shown in Figure 2, this means that the original 19-week schedule will actually take about 23 weeks.

Table 4. Tracking Progress with the Schedule Planning Template

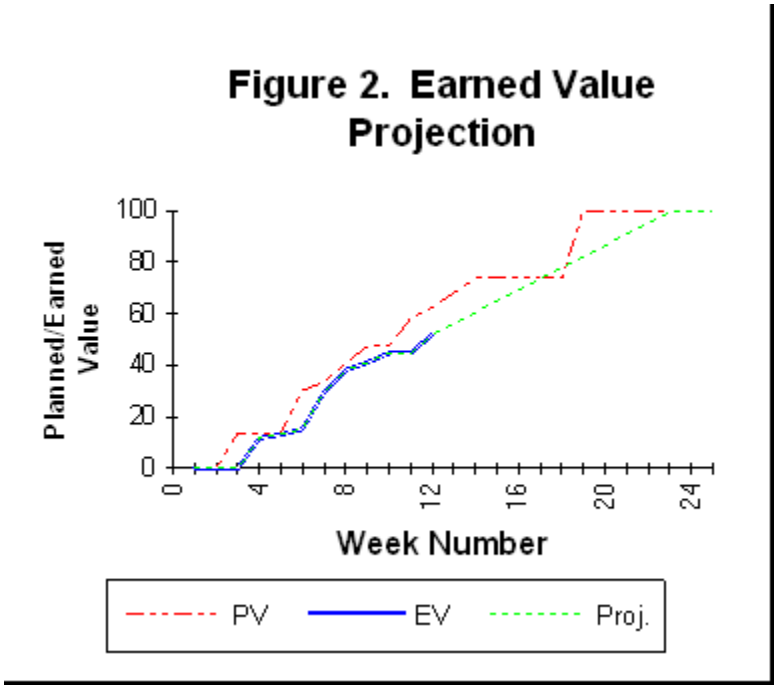
Week	**Planned Hours**		**Actual**	**Cumulative**	
Number	**Weekly**	**Cumulative**	**Hours**	**Plan Value**	**EV**

1	63	63	48	0	0
2	63	126	72	0	0
3	63	189	69	13.66	0
4	63	252	59	13.66	11.80
5	63	315	81	13.66	13.15
6	63	378	93	30.35	15.35
7	63	441	68	33.39	29.68
8	63	504	71	40.98	38.28
9	63	567	54	47.39	41.15
10	63	630	77	47.39	44.69
11	63	693	63	58.18	44.69
12	63	756	18	62.56	51.94
13	63	819		68.13	
14	63	882		73.86	
15	63	945		73.86	
16	63	1008		73.86	
17	63	1071		73.86	
18	63	1134		73.86	
19	63	1197		100.0	
20	63	1260			
Total	1260				

The earned value projection method can be quite accurate, as long as the project continues roughly as in the past. For example, if you added or removed staff, the earned value rate would change. Similarly, if you ran into unanticipated problems, you would probably be further delayed.

This is a key point. The earned value method is useful for estimating project completion as long as the original plan was reasonably sound and the work proceeded substantially as planned. It will not work, however, if the plan had major omissions. If you forgot to include system test, or allowed only 1 week when testing has historically taken two months, earned value will not provide useful estimates of the completion date.

The earned value completion estimate also assumes that there are no roadblocks. In this example, component 3 is 3 weeks behind schedule as of week 10. If it were to slip further, it could become the gating item on the entire project schedule. Thus, even though everything else progressed at the historical earned value rate, serious delays in component 3 could delay the entire project beyond the 23-week completion estimate.



Conclusion

The PROBE method shows you how to use historical data to make accurate estimates and plans. Once you have a good plan, the earned value method helps you track progress against it. While these earned value projections provide a clear picture of what is happening and an objective way to judge when a project will likely complete, they are not foolproof. It is important to examine the project and understand the causes of the deviations from the plan. The earned value method provides a useful yardstick for making these comparisons.

This concludes the discussion of the PROBE method for estimating with objects. You now have all the material you need to use these methods. You can apply these techniques to your personal work as well as to larger projects. While larger jobs will take more time to plan, the principles are identical.

Next month's topics

Next month we conclude this series on Estimating with Objects. That final column discusses the experiences of engineers who have used the PROBE method.

Thanks for your attention and please stay tuned in.

Watts S. Humphrey
watts@sei.cmu.edu