Understanding Software Project Estimates

Katherine Baxter
Champlain College

With two-thirds of software projects running long and over budget [1]*, it is important that upper management understand the value of proper estimation techniques, and that their estimators are as accurate as possible. This article discusses formal estimation techniques, accurate software estimation tools, the misinterpretation of estimation as target setting, and the accuracy of estimates.*

Software projects are notoriously complex and difficult to estimate accurately. Many authors have referred to estimation as a *black art*. Estimating a project accurately involves carefully analyzing data from many different aspects of the project, and using a number of different techniques to get the best estimate possible with the given information. Even then—depending on the accuracy of project information and how far along the project is in its life cycle—the estimate may still not be very close to the actual values at the project's completion.

Accurate project estimates, even early in a project's life cycle, are extremely important to an organization's success. For example, when prospective customers have received their requested proposals for a certain project and have gone through their initial proposal evaluation process, those *short-listed* organizations will have to submit cost and schedule estimates. If the organization awarded the project has initial estimates that are too optimistic, they may get stuck developing a project that runs over budget or breaks the contract in terms of scheduling; the end result may be losing revenue instead of turning a profit. If a certain organization's initial estimate is too pessimistic, they are likely to be rejected in favor of another whose estimates look more favorable to the customer. When so much is resting on finding accurate estimates—especially early in the requirements definition phase of a project—it is in every company's best interest to apply any and all available techniques to make sure their estimates are as close as possible to the actual values.

Despite many developments in estimation techniques, most project estimates are still not very good. In fact, only about one-third of all projects are completed on-time and on-budget, with some off in both areas and others so far gone that they are discontinued before completion. These numbers represent huge losses in profit. However, it is sometimes difficult to trace these losses directly back to problems in the estimating processes.

A number of different factors play into this problem of poor estimation. One of the most common is that, in many cases, accurate estimation techniques are simply not applied. Many organizations, in the interest of saving time and money, try guessing at project estimates with no formal process for determining the project's cost and scheduling needs [2]. This is most likely due to a lack of understanding of the importance of estimates. Estimating accurately involves a lot of time and money, and there is not always any direct and easy way to see a return on any investments made in proper estimation processes.

Sometimes organizations will try to take shortcuts to save money in estimation. One of the most common mistakes is basing estimates entirely off of historical data from past projects. The estimator will simply find a project that seems similar to the project they are estimating and use the final values from that project. While this technique is an important piece of the estimation process, when used alone it is a proven cause of schedule and cost overruns [2]. Since the industry is constantly developing and changing quickly, estimates based entirely on historical data are not enough. For accuracy, historical data should be used alongside other methods to find the most accurate possible estimates. It is important to use those values, but also to check them against values obtained from the latest updated estimation models and tools to keep up with the industry as it changes.

Another common problem is that deadlines are sometimes set before estimates are even made. The estimator then has to count backwards from the deadline in order to make the estimate instead of forming their own schedule [1]. This discourages a proper estimating process and is likely to influence the estimator into making overly optimistic estimates.

Even when good estimating models and tools are used, sometimes not enough effort is put into making sure the estimates are based on accurate or complete information. As one author puts it, "any estimate is only as accurate as its least accurate input variable" [1]. It is extremely important that estimators take the time to make sure all their data is as accurate as possible for that stage of the project's development before committing to any final estimates. If the data used to make the estimates is incomplete or inaccurate, the resulting estimate will also be incomplete and inaccurate.

There are a number of tactics that can be applied to help solve some of these problems. The most important of these is that a proper formal estimation process should be used in all cases. Guessing or taking shortcuts to come up with quick results will not provide the quality estimates that are needed. Formal estimation techniques are proven to make estimates much more accurate, especially early in project development when the requirements and risks are not yet clearly defined [2]. When so much rests on estimates being accurate, it is important not to cut any corners.

Another technique is to use multiple estimating techniques in combination. Using historical data from your own organization (as well as from others) and a few different models and tools can help find the best possible estimate [3]. Each of the estimates found from these techniques is an approximation. It is very unlikely that any of them will be exact: Multiple techniques can bring the final estimate closer to the right target.

Since all estimates are really only approximations, it is best to avoid settling on any single number. Early on—when requirements may not be completely defined and other changes in the project's completion plans may still take place—estimates are often significantly inaccurate. One technique to improve accuracy is to provide a best-case-scenario estimate, a worst-case-scenario estimate, and an expected estimate. The final estimate can then be provided as a range of values, rather than committing to one specific value from the start [1]. For example, an estimator may say that a project is likely to cost between $350,000 and $600,000. This gives the organization a general idea of how much the project will cost, but does

not fool them into thinking the estimate is more than an approximation.

For large-scale projects, however, manual estimation is not enough. There are a number of complex tools available to help create accurate estimates; for complicated projects with many parts, these are sometimes the only way to get a good estimate. In general, these tools are much more accurate than estimation by hand. Manual estimates tend to be too optimistic, often by more than 30 percent [4]. Even when they are done more carefully, a manual estimate will rarely be more accurate than an estimate made with the help of one of these tools.

Understanding the importance of accurate estimation—and a willingness to put in the resources needed to support good estimation processes—are vitally important to an organization's success. Learning what is involved in making good estimates, and what techniques can help improve estimate accuracy even more, are important first steps.

Estimation Tools

There are a number of software estimation tools available to help in making accurate estimates. For larger projects of more than a thousand function points, it is almost always necessary to use these tools rather than attempt manual estimation: They are complex and have so many factors that must be taken into account that an estimation tool is much faster than estimating by hand [4]. Even for smaller projects, there are a number of advantages to using automated tools rather than pure manual estimation.

Automated estimation tools tend to be much more accurate than manual estimation. Since there is less human interference, the estimates are less likely to be influenced by human bias that might make them unrealistically optimistic. Automated tools are also much less likely to underestimate or overlook effort involved in areas such as design, documentation, and testing [5].

There are many features that are fairly standard across most estimation tools that will assist in sizing estimates and estimating at the phase, activity, and task levels. They will also help with general quality and reliability estimation. Most will support size measurements in both function point and source lines of code units, and support conversions from one to the other. Most tools also have support for a variety of different languages, including older languages such as COBOL and FORTRAN.

Some tools also provide other features, but these are not standard across all tools. Some will perform risk and value analysis, inflation calculations for long-term projects, and currency conversions for international projects. Some will provide support for various standards like the SEI's CMMI®. Many tools also allow the organization to input historical data, which is then used to adjust scheduling and other estimates. Because estimation tools are often used in combination with project management tools such as Artemis or Microsoft Project, many also provide interfacing capabilities [4].

Even with all these features, there are still many aspects of a project that must be accounted for manually. They include things such as fees for trademark and copyright searches and legal expenses for any breach of contract if a project is not completed on-time or on-budget [4]. Generally though, when all input factors are carefully considered and are accurate, estimation tools will provide a thorough and accurate estimate of a much higher quality than can be achieved by hand.

"Understanding the importance of accurate estimation, and a willingness to put in the resources ... are vitally important to a company's success."

Estimation Isn't Target Setting

One common misunderstanding is that estimation is simply the process of finding a target end-date and total cost for a project. This assumption is dangerous because estimates are not exact, and early estimates are not nearly precise enough to pin down exact values. Committing to specific values early on is likely to cause the project to either go over schedule or run too long.

Estimates can help provide a general idea of when a project will be finished and how much it will cost, but it is impossible to settle on specific values with total certainty. For example, one could estimate that a project is likely to be finished sometime between 10 and 16 months, but there is still a chance that the project will not be completed within this time frame. Even setting a target end-date at the end of this range could be dangerous. Moving the target end-date with no particular purpose does nothing but change the probability of the project actually finishing on schedule [2]. Clearly, this estimate is not a good basis for choosing a specific end-date.

Target setting takes place when an external factor is determining a required end-date or budget. If a project must be completed by a certain event (like a specific conference) or by the end of a fiscal year, then a target end-date is set. The estimation process then becomes a matter of deciding how much can be completed by that date with a certain amount of resources, and refining the requirements to fit into this time frame [2]. Working backwards carefully like this can ensure that the project finishes by a certain date, but it may involve sacrifices in requirements or an increase in resources. If there is no specific end-date, using the estimation process to set one is risky.

Estimation is a process of taking a set of input values, conducting some careful analysis, and ending with a set of results. Once these results are found, it makes no sense to try to argue against or change results. The estimation process cannot be altered to achieve different results, and trying to change them to fit within a certain budget or time frame will only make the project less likely to succeed. If the results of an estimate are not satisfactory, the only reasonable way to change them is by adjusting the inputs—the information that the estimate was based on.

Inputs, like requirements and resources, determine the final results of estimation. Changing these values is the only way to reasonably affect the estimation results. If the estimated cost of a project is too high, it may be possible to adjust it by reducing the functionality of the system, consequently making the project smaller. If the estimated time frame is too long, it can often be changed by adding more resources to the project. These are the only meaningful ways to change estimated values.

Accuracy of Estimates

Early on in a project's development, estimates are likely to be very inaccurate. Estimates are based on inputs concerning what is known about the project; consequently, if the concept of the project is not entirely clear or does not match exactly with what the finished product will look like, the estimate is likely to be pretty far off from the final values. Functionality may need to be added or adjusted as the product is developed because require-

ments change often during project development and systems often don't work exactly as planned. Because this uncertainty is based in the inputs and not in the estimation process, spending more time on the estimate itself will not necessarily make it more accurate [6]. While spending extra time in the planning phase and ensuring all of the processes used are mature can help increase the accuracy of early estimates, it is still not accurate enough to meaningfully commit to specific values [2].

As the project progresses—and requirements become clearer and eventually *set in stone*—estimates become more accurate. Estimates in the conceptual phase (before extensive planning is done) are often off by as much as 50 percent, but are down to approximately 25 percent by the time functionality is determined. By the time the project is actually being implemented, estimates are usually within 10 percent of the final values [7]. It is important to note that these values are based on estimates made by skilled, experienced estimators and formal estimation methods. If the estimators were less experienced or used less precise estimating techniques, their estimates would likely be even further off the mark [6].

The most significant improvements in estimation accuracy occur during the first 20 to 30 percent of project development [6]. This represents the planning phases where the unknowns that cause such problems with estimation accuracy are being eliminated. To understand an estimate at any given point in development, it is important to understand how precise one can expect estimates to be at that stage.

Because early estimates are so inaccurate, it is important that they are never treated as exact expected final values. Early estimates should always be expressed in ways that clearly show this uncertainty, such as describing them as ranges of values rather than fixed points [2]. It is also essential that no commitments to specific values are made early on. Because there is so much possibility for inaccuracy early in development, any commitments made within the first 30 percent of a project's life cycle are not reasonable or meaningful [6].

Estimates should never be treated as exact final values, but early estimates in particular should be treated with care. As one author explains, "the only time we have sufficient data to truly warrant the label 'accurate' is at the very end of the project when all the variables are resolved. Unfortunately, no one will ever ask for an estimate at that stage" [8].◆

References

1. Dekkers, Carol A. "Creating Requirements-Based Estimates Before Requirements Are Complete." CROSSTALK Apr. 2005 <www.stsc.hill.af.mil/crosstalk/2005/04/0504Dekkers.html>.
2. Henry, David. "Software Estimation: Perfect Practice Makes Perfect." CROSSTALK June 2002 <www.stsc.hill.af.mil/crosstalk/2002/06/henry.html>.
3. Stutzke, Richard. "Software Estimation: Challenges and Research." CROSSTALK Apr. 2000 <www.stsc.hill.af.mil/crosstalk/2000/04/stutzke.html>.
4. Jones, Capers. "Software Cost Estimation in 2002." CROSSTALK June 2002 <www.stsc.hill.af.mil/crosstalk/2002/06/jones.html>.
5. Jones, Capers. "Software Cost Estimating Methods for Large Projects." CROSSTALK Apr. 2005 <www.stsc.hill.af.mil/crosstalk/2005/04/0504Jones.html>.
6. "The Cone of Uncertainty." Construx. 2008 <www.construx.com/Page.aspx?hid=1648>.
7. Roetzheim, William. "Estimating and Managing Project Scope for New Development." CROSSTALK Apr. 2005 <www.stsc.hill.af.mil/crosstalk/2005/04/0504Roetzheim.html>.
8. Armour, Phillip. "Ten Unmyths of Project Estimation." Communications of the ACM Vol. 45. No. 11, Nov. 2002.

About the Author

Katherine Baxter is a programmer at Champlain College's Emergent Media Center, where she creates computer games for use in education and training. Baxter also recently developed educational games for children at Tertl Studos in Montpelier, Vermont. She is completing her bachelor's degree in software engineering at Champlain College, where she will graduate this spring.

Champlain College
PO Box 670 Box 51
Burlington, VT, 05401
Phone: (978) 807-1793
E-mail: baxter.katherine@gmail.com