

## ESTIMATING PROCESS

*“Determining the size of system functionality and measuring the performance of project teams is the basis of successful projects.”*

Most people involved in IT projects have had to face challenges on meeting budgets, schedule, functionality and quality. Research (e.g. CHAOS Standish Report) shows that projects have difficulties meeting expectations. The organisation expects delivery of software that is working according to the defined requirements, is available at the agreed upon time, is within the stated budget and is of good quality. Meeting these expectations is not only important to the senior management, but also in particular for the business goals (time-to-market, achieving efficiency improvement) and satisfying the customers expectations (services and products are available at the agreed moment).

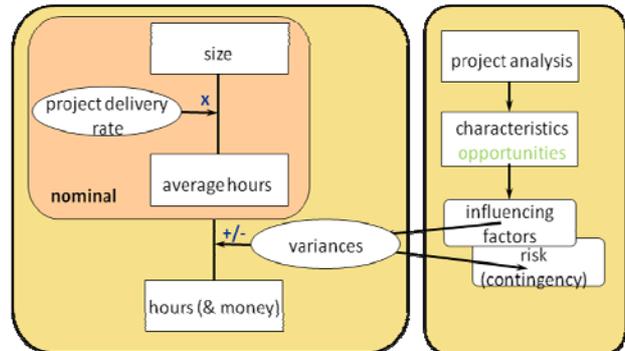
### Estimate

The first project delivery date and budget is set by the estimate. In most of the successful projects the use of a method to determine the size and the application of quantified historical data turns out to be the basis for managing expectations (schedule and budget) and providing the foundation for project management.

A consistent and standardised estimating process is the basis for achieving success. There are various models available, they may have slightly different setups but they are all using the same principals. One example is the 10-Step Process proposed by Dan Galorath<sup>1</sup>:

1. Establish scope
2. Establish technical background
3. Collect data
4. Estimate and validate size
5. Prepare baseline estimate
6. Review, verify and validate estimate
7. Quantify risks and Risk Analysis
8. Generate a Project Plan
9. Document estimate and lessons learned
10. Track project throughout development

For the purpose of this sheet the focus is on step 2 thru 8. This is best visualised in the enhanced Input-Process-Output (IPO) model.



### Enhanced Input-Process-Output model

The size [step 4] is the main driver for the estimate. The Project Delivery Rate (PDR) reflects the nominal performance within the technical background [step 2]. As performance (effort and cost) is related to size, this correlation is used to determine the base estimate [step 5]. The PDR is based on averages of historical data [step 3] of company projects, tool based data (e.g. SEER) or public data (International Software Benchmarking Standards Group – ISBSG). The baseline has to be reviewed and validated in respect of the specific circumstances of this project [step 6]. This project analysis is tuning the project looking at the specific variances to the nominal of product (e.g. complexity, reuse), process (organisation), people (skills, experience), technology (tools) and constraints (budget, schedule, capacity). This activity provides a refined estimate and an estimating uncertainty. The risk analysis [step 7] looks at the project from a different viewpoint, a more ‘What-if’ approach. The choices made to mitigate potential risks, can be divided in incorporated measures or actions when risks occurs. A common way is to include the impact of the pro-active measures as part of the variances in schedule and budget. The reactive actions are funded independently, from a contingency budget.

The final output is the Project Plan [step 8]. In addition to the budget and schedule, the plan gives an overview off the scope, purpose, methodologies, product, process, people, technology, constraints and risks. All assumptions, selections (nominals), decisions, risks and measures are made transparent. As much as possible quantified but at least qualified. The expectations are set and a confidence level associated.

<sup>1</sup> “Software Sizing, Estimation and Risk Management”, D. Galorath, Auerbach Publications 2006

## Sizing

Sizing is the key. Various methods for Functional Size Measurement (FSM) are available. The most used method worldwide is Function Point Analysis (FPA). The basis is data-driven mainframe software. Although updated over the years, the approach is not most suited for all development platforms (Object Oriented, Component Based), all architectures (real-time, multi tier) and all domains (embedded systems, avionics). A more recent FSM method COSMIC, might be better suited in more contemporary situations. In December 2002 both methods were certified by ISO (ISO 14143 compliant). These methods are explained in more detail in separate sheets available in the Software Estimation Series.

### *Feasibility Study and Measurement*

When requirements (high level) are defined, the software to be developed, rebuilt, enhanced or customised can be sized. In some circumstances it may make sense to use a proxy sizing approach rather than detailed sizing. By applying the IPO model this will provide a rough order of magnitude estimate (ROM) which can be used to manage initial expectations. The ROM is the basis either to request the appropriate budget or to re-scope the project to fit in given constraints. Focus is on having the functionality aligned with schedule and budget.

### *Development and Measurement*

Decisions on functionality have already been taken and requirements are more clearly defined. At this stage detailed functional sizing is more common. The estimate is now used to set clearer goals and to determine the required capacity. When data is available on labour on activity allocation, the required resources with appropriate skills can be scheduled correctly. The focus moves towards schedule and budget, less on functionality.

## Performance measurement

In addition to size, the performance is a crucial factor. Quantifying performance gives a clear insight into the effort and the costs per unit. Based on objective (quantitative) data instead of subjective feelings, suppliers, process improvement, tools, development languages and platforms can be compared.

The emphasis should be on: project delivery rate (size delivered / actual effort), speed of delivery (size delivered / actual elapsed time) and quality (number of defects in period / size). It's important to keep the people, process and technology factors in perspective.

### *Project Delivery Rate (PDR)*

At project phase transitions (milestones) the functional size covered by the deliverables is compared with the actual performance. For benchmarking, the data has to be normalised, taking into account the project specific characteristics. The model is now used 'bottom-up'.

## Tools

Although a tool is not required to implement an estimating process, it definitely supports the application of the process and the acceptance. The added value of a tool is described in another sheet in this series.

## The benefits at a glance

A standardised estimating process gives:

- insight in to functionality and size
- a realistic schedule and budget
- support to decision making
- support in establishing scope
- insight in to the performance
- a quantitative foundation for initiatives on software process improvement

## Galorath and Estimating

Estimating is the core business of Galorath. Over the years Galorath has gained a lot of experience in estimating. This is reflected in the SEER suite and the support organisation. Galorath and associated partners do have a standard approach, known as MOUSE, to embed the estimating process (with or without a tool) in the organisation and to train the employees.

## Like to know more?

We would like to tell you more about Estimating, SEER and related subjects. You can contact one of our consultants for a talk without any obligations. Just send an email to [pi@galorath.com](mailto:pi@galorath.com).

On [www.galorath.com](http://www.galorath.com) you will find a more extensive overview of the possibilities and services Galorath offers.

*The paper "Software Sizing, Cost, Schedule, and Risk: A 10-Step Process" can be downloaded for free from the Galorath website.*