Knowledge Based Project Duration Estimation for Workflow Based Document Management Software Projects

Marius VETRICI, Cristian IONIȚĂ
mariusvetrici@softmentor.ro, crionita@ie.ase.ro

This paper analyzes the existing types of duration estimation models for software projects. We focus on Metrix model and investigate its applicability for duration estimation of software projects that focus on delivering workflow based software systems. The Metrix model is a stochastic, knowledge-based Monte Carlo simulation over an activity graph. The model overcomes the limitations of other existing models by relying on a knowledge-base of precalculated discreet probability distributions for task duration. These probability distributions are automatically derived using the historical task duration estimation of the team members.
Keywords: *project, duration estimation, knowledge, workflow, document management*.

Introduction

In the last 10 years document management systems became more and more popular in the large and medium companies. Although the implementation of these systems brought an increase in business performance, they are far from reaching their full potential. The biggest problem associated with these systems is the integration of the system with the business processes they support.

The Document Management Platform

Document management systems are computer systems used to track and store electronic documents and images of paper documents. An example of such system is the DocuMentor platform created by Soft Mentor ([SOF08]).

The platform has all the standard features of a document management system. It can be used to define document metadata such as the hierarchical structure used to organize the documents and the data fields that need to be stored for each file. The client application can be used to index electronic documents or document images captured using a scanner according to the metadata definition. Document retrieval is performed based on full text indexes and / or document fields. Each document has an associated status that can be used to retrieve documents that are in different stages of processing.

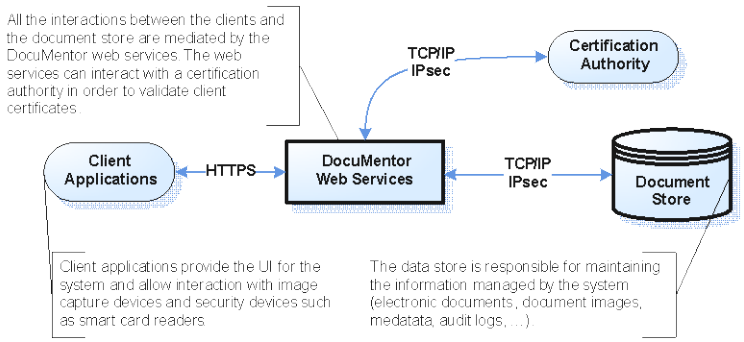

Fig.1. Architecture of the document management platform (simplified)

Business Process Automation Using Workflows

All modern businesses depend on complex business processes in order to conduct their daily activities. These processes involve documents, people and internal or external information systems [ION07].

The DocuMentor platform presented in the previous section is an example of process aware system. A process aware information system is a software system that manages and executes operational processes involving people, applications, and/or information sources on the basis of process models

([DUM05]). These models are typically instantiated multiple times and every instance is handled in a predefined way (possibly with variations).

One way to model business processes is using workflows. A workflow is a series of related interactions between computer systems or between people and computer systems. These processes modeled using workflows are executed using a workflow management system (WFMS). A WFMS is a software component that takes as input a formal description of business processes and maintains the state of a business processes executions, thereby delegating activities amongst people and applications.

Document Management System Implementation

The implementation of DocuMentor system for a specific client is a software development project in itself. That is because tailoring the document management platform to specific business needs involves several stages, like designing DocuMentor workflows according to company business processes, building electronic forms for data input and designing reports for data analysis and report. Even though we only customize the document management platform, the business demands can vary a lot both in size and complexity and thus, the software project too.

The grand majority of software development projects are known to be late and over the budget. Most of them hit schedule and budget overruns of 25% to 100% and sometimes even more [CHO07], [CHA06], [ROB01], [MCC96].

In order for the document management system implementation to be profitable for the end client, two criteria must be met:

a) The project should be delivered at the deadline.

b) The project should be delivered within budget.

This paper focuses on optimizing the estimation of the deadline of the project. The prerequisite for defining an accurate project delivery date is a precise estimation of the project duration.

The classification of duration estimation models

The range of duration estimation techniques and methods significantly broadened its coverage in the last years so that now we have sophisticated mathematical and statistical models and even expert system based estimation models.

Figure 2 depicts the classification of existing models [TEM07]:

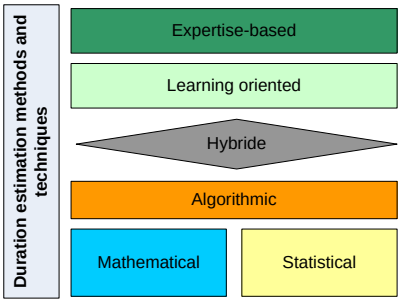

Fig.2. The classification of duration estimation models and techniques for software projects

Here under we generalize existing duration estimation methods for software projects together with their pros and cons:

- **Expertise-based methods**: These are the most flexible methods that can be easily adapted from project to project in order to enhance the quality of duration estimations. Meanwhile, they are too subjective. Depend on the experience of the experts in question.
- **Learning-oriented techniques**: Are based on real life examples that have been previously executed. Unfortunately the necessity to identify the key-variables is a daunting, time-consuming task because of the specifics of every project.
- **Algorithmic methods**: These methods are able to refine their estimates on subsequent iterative algorithm execution. Can be easily adapted to the variations of the input values. Unfortunately the estimations can have a very low quality when the input data has not been properly validated and calibrated.
- **Mathematical-statistical models**: Are easy to develop and have a very good academic background. Still these models need a large set of historical data.
- **Hybrid methods**: Are the most efficient

by combining key aspects from all other methods. Yet these models are immature, undeveloped and lack solid formalization.

The Metrix model for software project duration estimation

In order to overcome the disadvantages of the existing duration estimation models described in previous section of this paper we will used the Metrix model proposed in [VET08]. This is a knowledge based, stochastic model that addresses the project duration uncertainty by running Monte Carlo simulations over the activity graph.

The advantage of this approach is that the model produces an interval for the possible project durations and a probability distribution. Thus, one is able to know the possible project durations together with the probability that certain duration will materialize. As follows, the Metrix model structure and the steps it encompasses are presented in greater detail.

The knowledge-base containing the history of the duration estimations for the tasks that have already been finished represent the input data of the model. The result of running the model is a probabilistic distribution of the project duration. The steps performed are described here under:

Step 1. The historical task duration estimations are collected for every developer. Will be considered both current project finished tasks and the tasks finished in other projects during the last 6 months.

Step 2. For every historical task duration estimation from step 1 we calculate the Estimation Accuracy Index (EAI) using the following formula:

$$EAI = \frac{ED}{AD} (1)$$

where: ED–estimated task duration (in hours); AD–actual, elapsed task duration.

Using the results above we calculate the discreet probability distribution for the EAI indexes for every developer part of the team.

Step 3. On build the activity graph using the task dependency and estimated task durations.

Step 4. On find the critical path through the graph and on calculate the deterministic duration of the software project.

Step 5. On run the Monte Carlo simulation. The following operations are performed at each stage:

a) for every task, on adjust the estimated duration with a randomly chosen EAI (using the probability distribution from step 2).

b) on recalculate the critical path method and the project duration.

On repeat the simulation between 1000 and 10000 times.

Step 5. We calculate the project duration frequencies obtained as a result of Monte Carlo simulation. We display the project duration probability distribution. See figure 3:

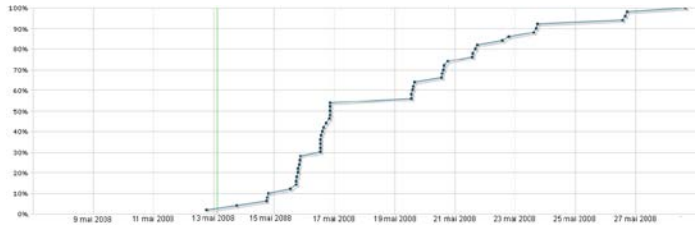

Fig.3. Probability distribution for project duration and project deadline

Conclusion

The first benefit of the Metrix model is that unlike classical deterministic models, which offer a single value for the estimated project duration, this model produces a probability distribution of the software project duration. By using this approach we reduce the project uncertainty by allowing the manager to gain better control over the project duration and the associated probability of a certain duration outcome.

The second benefit of the Metrix model is

that it relies on the historic duration estimation of the team members. Similar models based on Monte Carlo simulations require a duration probability distribution function for every task. This requirement unfortunately set Monte Carlo simulations out of the practical domain into the academic universe. The innovation brought by the Metrix model is the elimination of the probability distribution functions requirement and the use of discreet probability distribution of the EAI (defined in this paper). The EAI probability distribution can be easily determined using the historical estimation errors which are at the disposal of most software companies.

Bibliography

[CHA06] ***, *The Chaos Report of IT Project Failure*, Standish Group, 2006

[CHO07] A. W. Chow, B. D. Goodman, J. W. Rooney, C. D. Wyble, *Engaging a corporate community to manage technology and embrace innovation*, IBM Systems Journal, Vol. 46, No. 4, 2007.

[DUM05] Dumas M, van der Aalst W, Hofstede A, *Process-aware Information Systems*, John Wiley & Sons, 2005

[ION07] C. Ioniţă, *A Domain Specific Language for Secure Document Management*, Conferinţa Internaţională de Informatică Economică 2007

[MCC96] Steve McConnell, *Rapid Development*, Microsoft Press, Washington, 1996

[ROB01] ***, *ERP Software Implementation Success Rates*, Robbins-Gioia 2001

[SOF08] ***, http://www.softmentor.ro/, Soft Mentor Website, 2008

[TEM07] Temnenco V., *Software Estimation, Enterprise-Wide*, IBM The Rational Edge, Vol. June 2007

[VET08] Vetrici M., *Software Project Duration Estimation Using Metrix Model*, Revista Informatică Economică, Vol XII/Nr. 2/2008