

# Broadleaf

**Creating value from uncertainty**

Broadleaf Capital International Pty Ltd

ABN 24 054 021 117

[www.Broadleaf.com.au](http://www.Broadleaf.com.au)

## Resource:

# Quantifying the risk to software project costs

In principle, software project cost risk has the same characteristics as the cost risk in other types of projects. However, there are good reasons why software projects have a reputation for being prone to over running budgets and schedules. This note outlines some of the factors involved and how they can be used to make realistic assessments of cost risk, achievable targets and contingency requirements.

Version 1, 2017

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Cost and risk in software projects</b>	<b>3</b>
2.1	Context	3
2.2	Common sources of uncertainty	6
<b>3</b>	<b>Modelling cost uncertainty</b>	<b>8</b>
<b>4</b>	<b>Conclusion</b>	<b>12</b>
<b>5</b>	<b>Contacts</b>	<b>13</b>

## Tables

Table 1: Range analysis table	10
-------------------------------	----

## Figures

Figure 1: Example context	4
Figure 2: Disruptive forces	5
Figure 3: Cost drivers	9
Figure 4: Risk modelling relationship	9
Figure 5: Sequence of assessing values	11
Figure 6: Cumulative distribution	11

## 1 Introduction

Software projects have a reputation for being difficult to forecast and control. Agile development can help to ensure that costs do not exceed a limit by working to deliver as much value as possible within a defined budget. The final scope delivered cannot be guaranteed but financial control may be easier.

Many straightforward software projects are still implemented using traditional methods, loosely referred to as the waterfall approach. Projects delivered in this way are constrained to meet a defined scope. They place stress on owners and developers if the available funds are exhausted before the scope is complete.

The use of an agile, waterfall or a hybrid approach is secondary to deciding whether to embark on a development in the first place and how much funding to provide. The path taken by an agile development, and even what it ultimately will deliver, might be adjusted as the work is underway. However, there must still be some sense that the funding available is commensurate with the capability that is expected to be delivered.

Methods used to analyse cost uncertainty in other kinds of projects are applicable to software projects. However, some of the features of software projects affect the emphasis given to various parts of the analysis, and it is useful to be clear about what these are and how we can handle them.

This note cannot cover every aspect of software project cost risk. It outlines some of the common issues that affect software projects and illustrates how they can be included in an analysis.

## 2 Cost and risk in software projects

### 2.1 Context

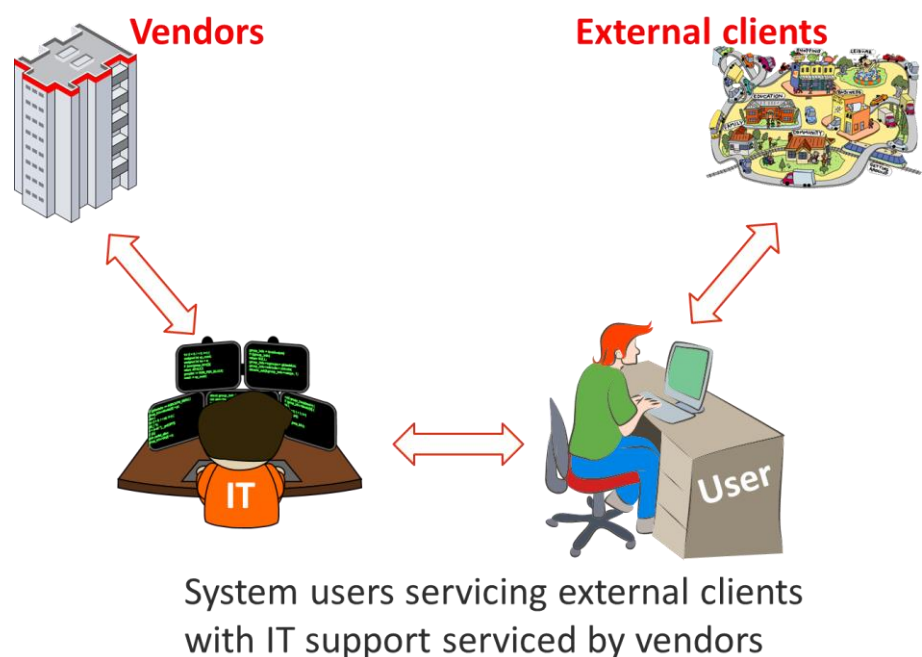
The ways in which software projects differ from work such as engineering or infrastructure asset construction stem from two key features of software development:

- Software projects deliver tools, knowledge and information rather than tangible material assets
- The proportion of the total cost associated with human effort is generally far higher in software projects.

# Broadleaf

Despite having been surrounded by and dependent on software for decades, the fact that software deliverables are intangible still colours the way software projects are managed. Senior stakeholders are more inclined to request disruptive changes to a software development than they are to suggest rerouting a freeway or relocating a process plant once construction work has started. Some of the forces at work are illustrated in Figure 1.

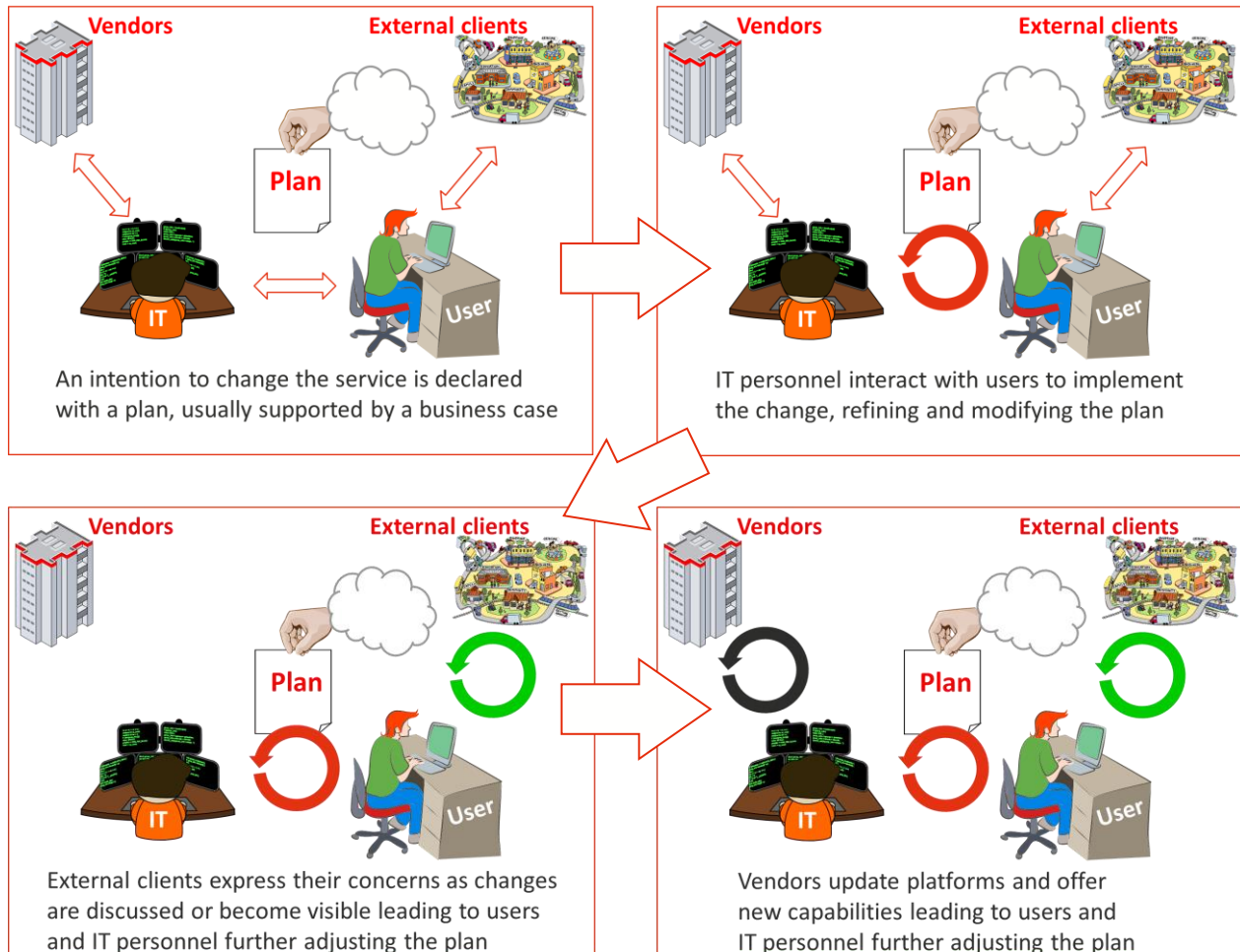
**Figure 1: Example context**



This system might operate smoothly until a project is established to bring about change. Ways that seemingly innocuous and quite legitimate interactions can create challenges for such a project are illustrated in Figure 2.

# Broadleaf

Figure 2: Disruptive forces



Not shown in this simple illustration are further sources of complexity, such as the existence of multiple centres of power within the organisation each able to exert an independent influence on the work. On a wider front, the services being delivered may span multiple organisations, requiring co-ordination between another set of loosely aligned but independent stakeholders.

All these processes can change the priorities driving a system development more quickly than the development can be completed. For more on this see resources dealing with [complexity in software procurement](#).

Regardless of the development approach that will be adopted, organisations need to assess the level of funding that will be required to deliver the capability they need. Details of how that funding will be expended might evolve as the work progresses but there needs to be a rationale for releasing funds and

allowing work to start. An estimate of costs is required and that estimate will be subject to uncertainty.

The following section is not an exhaustive description of all the sources of uncertainty affecting software project costs. It focuses on matters more likely to affect software projects than projects with tangible deliverables, which have been discussed in several [earlier papers](#).

## 2.2 Common sources of uncertainty

### Market rates

As with any project, there is often uncertainty about market rates for goods or services, including professional effort. Budgetary quotations provided by suppliers may prove to have been optimistic. The supplier might not have disclosed constraints associated with the rates they have provided or might not have devoted sufficient time to ensuring that their initial response was comprehensive. Conversely, if they feel confident of winning the business, they might have simply provided standard rates from a price list without considering discounts or other arrangements that could offer a saving.

### Scale and scope

Forecasting the demand that will fall on an IT system is difficult in all but the simplest of situations. At some point in the operation of most IT systems, human interactions play a role. Human behaviour can rarely be understood with great certainty, especially in a fluid social and administrative setting. Behavioural factors might affect the level of demand within the planned deployment of a system or lead to pressure to extend the deployment into new areas.

System demand can affect the cost of IT infrastructure, support services, licenses and other charges. It is not unknown for a system to be the victim of its own success and experience stress from higher than expected levels of usage. If this becomes apparent after a project is completed, the costs will fall elsewhere, but if additional demand is foreshadowed during the project implementation then the project budget may have to fund the increased provision.

## Effort

Any knowledge-based work is subject to uncertainty about the size of the task to be completed. This is true of the preparation of designs for physical infrastructure as it is for almost all stages of software projects from requirements capture to acceptance testing.

Even if the scale of the overall task to be undertaken is clear, IT projects often require work to be carried out by related entities who are responsible for systems that interact with the one that is being developed or enhanced. Project budgets might assume that these peripheral or interfacing tasks will be funded as part of the system maintenance activities of those entities or by their own project funding. It is good practice to capture the cost of all these additional activities in the overall business case for a project, but this is often overlooked. Even if all the costs are captured, control of the funding for work outside the core project often rests with the organisations or business units who are to carry it out.

Short-term pressures, difficult working relationships or divergent priorities can result in these additional tasks being delivered late or not at all. In many instances, the only way to make progress is for the core project to fund the work directly and it might not be possible to recover that cost from those who were to have carried it out. The way work is allocated between stakeholders can be a significant source of cost uncertainty.

Even if the amount of work to be carried out is well established, the productivity of the personnel undertaking the work may be uncertain. A team that is thoroughly familiar with the end user environment and IT platforms to be used will make faster progress than one that is not, and a team that suffers significant staff turnover will not be as productive as one that is stable.

Finally, the resources devoted to project management and administration are usually based on an expectation of the complexity and challenges the project will face. If the work is easier to manage than expected, it might not be necessary to staff up to the planned project management headcount or it might be possible to manage with fewer highly skilled or less expensive staff. Projects that are intrinsically more complex than anticipated, or that run into difficulties in any of the other areas mentioned here, could find they need additional personnel or more highly skilled and experienced personnel than anticipated. Such variations can manifest themselves as a change in the level of effort or as a change in rates.

## Scheduling

An IT project team's headcount can rarely be turned up and down at will in response to emerging patterns of over or under achievement on the schedule. The usual pattern is for the headcount profile to be more or less fixed at the level decided during early planning, so variations in scale, scope, productivity and work sharing show up directly as schedule variations.

Schedule variations can also arise due to external dependencies on supplier deliveries or regulatory approvals. Internal dependencies with implications for the schedule include the need to obtain approval to proceed, sign-offs on specifications or plans, the provision of specialist resources or the release of funding at key stages in the project.

Schedule variations drive variations in overheads such as the costs of the project management team, office space and development or testing facilities. They can also be linked to costs associated with sustaining existing systems that are to be retired when the project is complete. Maintenance contracts and license charges that were expected to cease at a specific time may need to be extended as the project continues past its planned completion date.

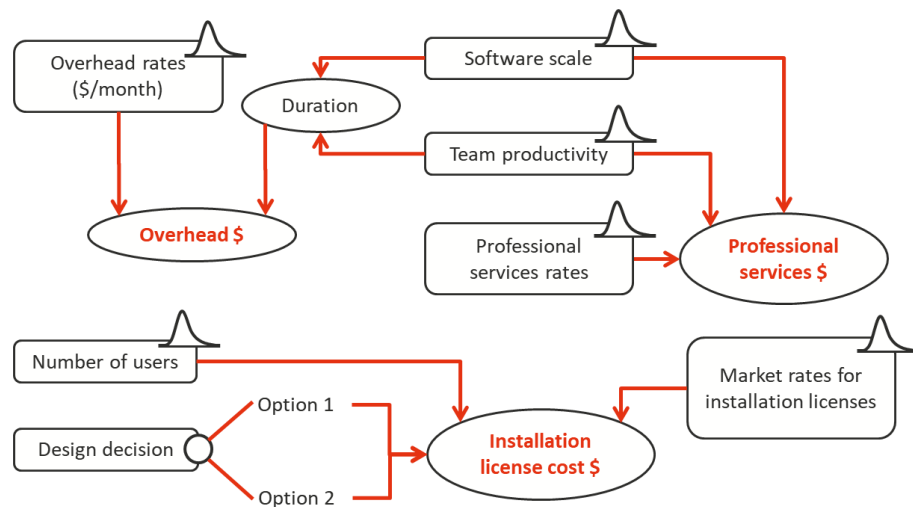
## 3 Modelling cost uncertainty

The bulk of most IT project budgets is associated with professional effort. As outlined in the previous section, direct development effort can be a driver of the schedule because progress is linked to the amount of work to be completed and the productivity of the personnel carrying it out. At the same time, the schedule can be a driver of management and support costs as any change in a project's schedule affects the duration over which these costs are accumulated.

The interaction between effort and duration is often the dominant process in an IT project's cost risk. This is illustrated in Figure 3. The scale of the project and the productivity of the development team drive the project duration. In turn, the duration drives the cost of overheads. At the same time, the scale of the project and the productivity of the development team also drive the direct cost of professional development effort. The diagram also illustrates interactions between market rates and other features of a project.



**Figure 3: Cost drivers**



A simple model for a small project might be no more complicated than shown in Figure 3. It would typically be constructed in a straightforward Monte Carlo simulation environment such as @RISK, ModelRisk or others.

Each of the items marked with a probability distribution icon would be represented in a model by a distribution function. The outcome of the design decision, such as a choice of deployment platform, would be represented by the probability of each option being selected and the costs associated with each option.

Where factors are combined, simple algebraic relationships can be used to represent their net effect. This is illustrated in Figure 4 for the simulated professional services (PS) cost. The  $\Delta$  terms represent the relative variation in the factors shown in their subscripts and  $PS_0$  represents the initial estimated professional services cost.

**Figure 4: Risk modelling relationship**

$$PS = PS_0 \cdot (1 + \Delta_{scale}) \cdot (1 + \Delta_{productivity}) \cdot (1 + \Delta_{PS\ rate})$$

Each of the distributions in Figure 3 represents the combined effect of the multitude of small factors that can influence the actual value of the quantities used to make an initial estimate of the cost. It is not necessary to extract from the estimate any specific values of the productivity or scale of the job. All that is required is to consider by how much any variations in productivity, scale or other factors could alter the costs that they affect. This is a judgement that

estimators and subject matter experts are able to make. These relationships are readily represented in spreadsheet based modelling environments such as @RISK and ModelRisk.

There is a well-established method for assessing and quantifying uncertain values, illustrated in Table 1 and Figure 5. This way of gathering information helps to minimise optimism and anchoring bias that can lead to unrealistic assessments.

**Table 1: Range analysis table**

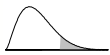
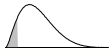

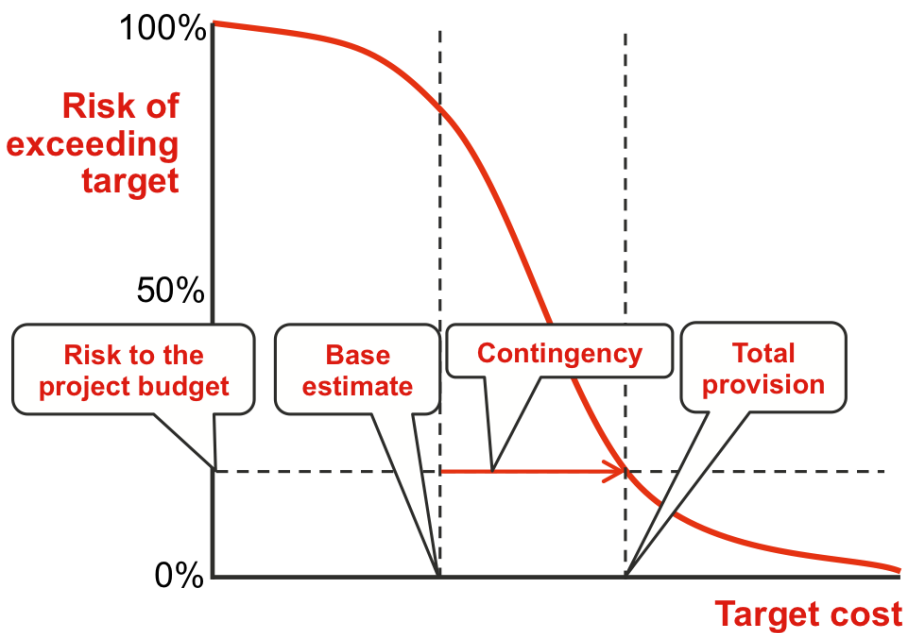
<b>Data table 1: Title or factor name</b>		
<b>Date</b>	<b>Participants</b>	
<b>Scope &amp; Assumptions</b>		
<i>Brief note of key assumptions underpinning the current plan and estimate in relation to this factor</i>		
<b>Status of work to prepare estimate and plan</b>		
<i>Summary of how this part of the estimate or plan was prepared (detailed bottom up analysis, historical data, factored, lump sum judgement call ...)</i>		
<b>Sources of uncertainty</b>		
<i>Brief description of the reasons this factor is uncertain</i>		
<b>Pessimistic scenario description</b>		
<i>Brief description of the circumstances we would see if it was going badly</i>		
<b>Optimistic scenario description</b>		
<i>Brief description of the circumstances we would see if it was going well</i>		
<b>Likely scenario description</b>		
<i>Where it is useful to do so, brief description of the most likely scenario</i>		
<b>Range estimate</b>		
Scenario	Forecast	Notes
Worst		<i>The most remote possible poor outcome, unlikely as it may be – a nightmare</i>
Best		<i>The most remote possible good outcome, unlikely as it may be – perfect textbook case</i>
Pessimistic (1/10)		<i>A pessimistic outcome that you feel has roughly a 1 in 10 chance of arising or being exceeded</i>
Optimistic (1/10)		<i>An optimistic outcome that you feel has roughly a 1 in 10 chance of arising or being bettered</i>
Likely		<i>The most likely outcome</i>

Figure 5: Sequence of assessing values



The characteristic output of a Monte Carlo simulation cost risk assessment is illustrated in Figure 6. This shows a cumulative distribution of the possible cost of a project and provides a means by which decision makers can set a contingency amount that will provide an agreed level of confidence that the project will have sufficient funding.

Figure 6: Cumulative distribution



## 4 Conclusion

Software project cost risk has the same underlying characteristics as cost risk in projects from other sectors. While human resources and effort are an important part of most construction, infrastructure and heavy engineering projects, they dominate software projects.

Schedule risk interacts strongly with cost risk. The schedule is affected by uncertainty about the amount of work to be done and the productivity a team can achieve, which also affect direct costs. In turn, duration uncertainty affects overhead costs and sometimes the cost of maintaining legacy systems.

For small projects, unless their timescale is critical, the interaction between effort, the schedule and overheads can often be represented realistically in a relatively compact model.

The keys to success in software cost risk analysis are to craft a model that links uncertainty to the cost realistically and to avoid bias in the assessment of modelling inputs. These issues are discussed in more detail in other [Broadleaf publications](#) including [one that focuses on how models are structured](#).

### Software risk summary

While not a complete list of the risk drivers encountered in software projects, the sources of uncertainty that have been discussed here are a useful starting point for thinking about software project cost risk:

- Market rates for professional effort, licenses, hardware and facilities
- Scale and scope of the deliverables
- Development team productivity
- Work split between the core project and related entities
- Project management team size and running cost
- Schedule duration variation driven by progress of the development team, external and internal dependencies.

## 5 Contacts

If you would like further information about this topic please contact us. We will endeavour to reply promptly.

**Dr Stephen Grey**

Grey@Broadleaf.com.au

**Dr Dale F Cooper**

Cooper@Broadleaf.com.au

**Pauline Bosnich**

Bosnich@Broadleaf.com.au

**Grant Purdy**

Purdy@Broadleaf.com.au

**Geoff Raymond**

Raymond@Broadleaf.com.au

**Phil Walker**

Walker@Broadleaf.com.au

**Mike Wood**

Wood@Broadleaf.co.nz

For further information, visit [www.Broadleaf.com.au](http://www.Broadleaf.com.au)