Decision Support for Planning Software Evolution with Risk Management

D. Greer
Queen's University Belfast
Des.Greer@qub.ac.uk

Abstract. Software evolution planning involves a decision making process about which changes should be introduced and when. This process is informed by knowledge relating to the existing product and its environment. There will be new enhancements as well as corrections to known problems. There is also knowledge of operational risks in the system that could manifest as problems at some time in the future. To counter these, risk reduction actions may be recommended. These can be treated as candidate system changes, with their own costs and possibly extra benefits. Any system changes will also carry with them development risks. To mitigate these further risk reduction actions may be formulated, again with an associated cost. However, there is typically a limited budget for the next release of the software. This constraint, along with that of inter-dependencies between candidate changes and taking account of the stakeholders varying attitude to risk means that there is a complex decision to be made. Using the fact that systems changes are estimable in terms of cost and benefit and that risks are estimable in terms of probability and potential impact cost, an approach is described that combines software evolution with risk management, incorporating simulation and genetic algorithms to support the decision making process. The method links product and project management and provides support for those involved in planning system evolution and is evaluated here using a sample project.

1. Introduction

System evolution is an important part of the software life cycle. This evolution can be *corrective* where already logged errors are fixed, *adaptive* where a program is enhanced to meet new requirements, *perfective* where performance is improved or *preventive* where an attempt is made to correct potential faults [6]. Despite the potential disruption of software failure, preventive maintenance receives less attention. Preventative maintenance accounts for typically 5% of maintenance activities, according to one empirical study [7]. One of the reasons for this might be an apparent oversight of the need for risk management relating to the software maintenance phase [1]. Hence, there is a need to ensure that risk management is performed as part of the software evolution process, including an assessment of *operational* risks in the existing system and *development* risks associated with any planned changes. Indeed there is an extra sensitivity to both types of risks in a maintenance situation due to the reduced opportunity for contingency planning (due to the constraint of having a legacy system) and the increased impact of risks occurring (due to a larger user community to possibly suffer) [1].

Where risks are identified and considered serious enough, risk reduction actions must be considered. For example, an operational risk could be that some part of the software may fail under a certain set of conditions. In this case a risk reduction action could be to amend the software to avoid or handle the fault. A development risk could be that the building of some component takes longer than expected. This might be mitigated by building a prototype to test the concept. In both cases there is an associated cost in making the risk reduction.

Current approaches to planning system evolution, typically involve the change manager distributing their budget among bug fixes and identified enhancements based on subjective benefit estimates and without the knowledge of the real cost of these and of potential problems that may be present as risks. However, in planning the evolution of an existing software system, the system planner must consider, alongside estimated benefits, the costs of i) enhancements; ii) corrections of known problems; iii) risk reduction for identified operational risks and iv) risk reduction for development risks associated with planned system changes.

The next section will describe the problem more formally. Section three will describe the proposed solution to the problem. Section four will describe a sample project and will illustrate the proposed evolution process. In the final section, some discussion on the results and indication of future research is provided.

2. Problem Description

i) After the review stage for an existing system there is a set of candidate changes, C. These changes have initially been established from suggested enhancements and corrections to the existing system. Each change is described by a textual description, t, a cost value, e and a benefit value, b.

$C = \{(t_1,e_1,b_1), (t_2,e_2,b_2), \ldots (t_n,e_n,b_n)\}$

ii) Also in the existing system, a set of operational risks O can be identified. These are represented by triplet of user-assigned values consisting of a description o_i, probability value, p_i and impact value i_i.

$O = \{(o_1,p_1,i_1), (o_2,p_2,i_2), \ldots (o_n,p_n,i_n)\}$, where $0 \le pi \le 1$

iii) Considering this set of risks may reveal the need for risk reduction actions, A, which may also be treated as changes. Thus, the set of changes is extended to C′

$C' = A \cup C$

(iii) Each change in C′ can alter the probability or impact values of any of the identified operational risks in O. Thus, we define a function Φ_C

Φ_C: $(o_i,p_i,i_{t1}) \rightarrow (o,p_i{}',i_i{}')$ where $p_i{}'$ is new probability after change C and $i_i{}'$ is the new impact value after change C.

(v) Changes may also introduce new risks. Thus for change C_i there is a new set of risks, X. Should that change occur the set of operational risks is expanded to O′ where $O' = O \cup X$

(vi) Each change in C′ may have associated with it a set of development Risks, D. This set consists of the quadruplet: the identity of the change c triggering the risk, the description of the risk, d, the probability, p of the risk occurring and the impact i of the risk occurring.

$D = \{(c_1,d_1,p_1,i_1), (c_2,d_2,p_2,i_2), \ldots (c_n,d_n,p_n,i_n)\}$, where $c_i \varepsilon C$, $0 \le pi \le 1$

(vii) Further changes, M can be defined to reduce development risks. These contain the identity of the development risk d, a description t, a cost e and an additional benefit b.

$M = \{(d_1,t_1,e_1,b_1), (d_1,t_2,e_2,b_2), \ldots (d_1,t_n,e_n,b_n)\}$ where $d_i \varepsilon D$

Thus, $C'' = M \cup C'$, $D' = \{(c_1,d_1,p_1',i_1'), (c_2,d_2,p_2',i_2'), \ldots (c_n,d_n,p_n',i_n')\}$

(viii) We will eventually choose a subset C^* from C″, but there may be cases of mutual exclusion. Therefore, we define a binary relation μ on C″ such that $(x_i,x_j) \in \mu$ implies $((x_i \varepsilon C^*) \wedge (x_j \varepsilon C^*)) = \text{FALSE}$

(ix) There may also be cases where candidate changes should be selected together or not at all. Therefore, we define a binary relation ν on C″ such that $(x_i,x_j) \in \nu$ implies $(xi \in C^*)$ XOR $(x_j \in C^*) = \text{FALSE}$

x) Suppose a function, χ returns the mean cost of a set of changes and a function, β returns the mean benefit accrued from those changes and that there is a user-set limit of L on the mean cost, then a set C* can be selected from C″.

$C^* \subseteq C'' : \chi(C^*) < L, [\chi(C^*) - \beta(C^*)]$ is maximised

xi) In any product there may be an operational risk exposure level that is acceptable. This is represented by the probability that the mean impact cost (ι) from a simulation using the remaining operational risks in O′ is equal to or exceeds the acceptable probability, p, that a set impact cost (γ) is incurred.

$P(\iota \le \gamma)) \le p$

xii) There may also be a development risk exposure level that is acceptable. This is represented by the probability that the mean impact cost (φ) from a simulation using the remaining development risks in D′ is equal to or exceeds the acceptable probability, q, that a set impact cost (κ) is incurred. $P(\varphi \le \kappa)) \le q$

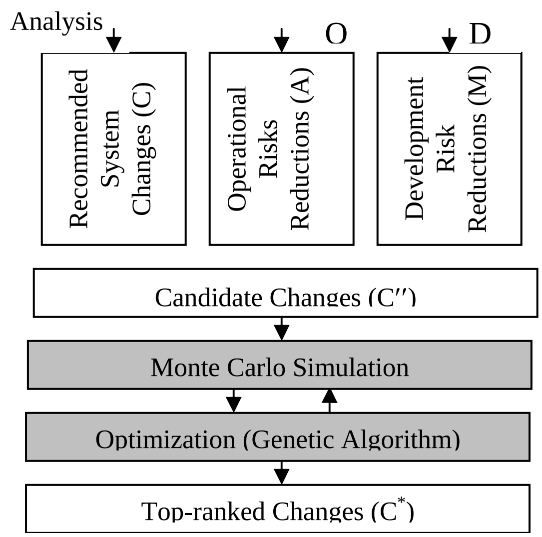

Figure 1: Decision Support Process

3. Decision Support Solution

Figure 1 describes the approach taken to supporting system evolution decision support. The process starts with an analysis of the current system. This can be by traditional methods, although in previous work, a goal oriented approach where the existing system is compared against an ideal model of the system, has proved to be particularly effective at revealing shortcomings and also suitable for identifying operational risks in the current product [4]. This analysis reveals required system changes (C) and operational risks (O). Consideration of these risks can induce further system changes that will reduce these risks (A). These are added to the system changes to produce a new set of candidate changes (C′). These in turn, are considered for possible development risk (D). Here it is possible to use risk questionnaires such as those from the SEI [3]. From these, further risk reductions may be established (M) and added to form a updated set of candidate changes (C″). The probabilities for the risks are used to create a discrete probability distribution of costs using Monte Carlo simulation. At the end of each Monte Carlo simulation, the genetic algorithm executes. Together the simulation and the genetic algorithm select changes from the candidate changes to form a number of solutions (C^*).

Genetic algorithms have been shown to be appropriate in situations where there are complex relationships between several factors and consequently a very large solution space [2] and indeed have already been successfully employed in similar problems [5].

4. Case Study

In order to illustrate the method, a sample system will be used, where 10 operational risks have been identified by risk analysts. Table 1 provides the data for these, the risks being labelled OR_n and each risk having with it a probability of occurrence over the lifetime of the system and an estimated impact in cash values.

Table 1: Operational risks for existing system in sample project

Operational Risk	Probability	Impact (000's)
OR_1	0.1	8
OR_2	0.5	7
OR_3	0.2	8
OR_4	0.4	14
OR_5	0.6	16
OR_6	0.7	11.5
OR_7	0.9	11
OR_8	0.1	10.5
OR_9	0.4	17.5
OR_{10}	0.3	6.5

As a result of analysis, considering existing problems and opportunities for innovation a number of system changes have been identified as shown in Table 2. Further, consideration of the operational risks induces additional risk reduction changes, as shown in table 3. These have the effect of reducing the risk exposure due to the risk by either reducing the probability of the risk or the impact of the risk, or both. In Table 3, C_{11}–C_{15} have been added to reduce risks OR_5-OR_9. In most cases there is no direct benefit from the change, but to illustrate the flexibility of our approach, one change (C_{14}) accrues a small benefit if implemented. In addition, table 4 shows that some of the risks are addressed by already defined by system changes. To mimic the real-world possibilities, in some cases a combination of changes is required to achieve the risk reduction (e.g. C_1+C_2 to reduce OR_1).

Table 2: System Changes with Costs and Benefits

Change	Cost ($000's)	Benefit $(000's)
C_1	16	56
C_2	17	54
C_3	16	102
C_4	19	72
C_5	15	86
C_6	8	67
C_7	23	97
C_8	10	36
C_9	17	46
C_{10}	14	94

Table 3: Operational Risk Reductions

Change	Cost (000's)	Benefit (000's)	Risk Reduced	New Prob-ability	New Impact (000's)
C_{11}	8	0	OR_5	0	0
C_{12}	7	0	OR_6	0.1	5
C_{13}	6.5	0	OR_7	0.2	2
C_{14}	9	4	OR_8	0	0
C_{15}	5.5	0	OR_9	0	0

Table 4: Risk reductions via system changes

Change	Risk Reduced	New Probability	New Impact (000's)
C_1+C_2	OR_1	0	0
C_2	OR_1	0.1	3
C_6+C_7	OR_{10}	0	0
C_{14}	OR_{10}	0.2	4
C_2	OR_2	0.4	2
C_2	OR_3	0.1	4
C_3	OR_4	0.1	14

While clearly not a desirable effect, the introduction of new operational risks as a result of the some system changes may occur. This is illustrated in Table 5.

Table 5: Operational Risks Introduced by Changes

Risk	Change	Probability	Impact (000's)
OR_{11}	C_{15}	*0.*3	5
OR_{12}	C_2	*0.*4	4

A further factor is that system changes may have risks associated with the development process. Typically these are risks relating to budget or schedule and may include personnel risks, technical uncertainties, changing requirements, and so on. These development risks for the sample project are illustrated in Table 6.

For each development risk, there is also the possibility of one or more risk reduction actions. These potentially reduce the probability and/or impact of the risk (Table 7) but have a cost associated with them.

To represent inherent dependencies, pairs of system changes were deemed to be mutually exclusive:

$$\mu=\{(C2,C3),(C4,C6)\} \quad (1)$$

Similarly, the following pair was deemed to be coupled:

$$\nu=\{(C5,C7),(C7,C10)\} \quad (2)$$

Table 6: Development Risk Assessment

Development Risk	Change	Probability	Impact (000's)
DR_1	C_2	0.3	5
DR_2	C_2	0.5	5
DR_3	C_4	0.4	16
DR_4	C_4	0.5	5
DR_5	C_6	0.6	2
DR_6	C_6	0.8	5
DR_7	C_7	0.5	10
DR_8	C_8	0.9	4
DR_9	C_8	0.7	3
DR_{10}	C_9	0.2	10
DR_{11}	C_{10}	0.1	4
DR_{12}	C_{10}	0.2	10

The overall mean budget for the sample project was set at 130K, so that no solution was allowed that exceeded this. A further constraint was in the levels of risk exposure in the proposed system and in the development process. For operational risks, the constraint was set that there should not be greater than a 20% chance of a 10K impact. For development risk, a 10% chance of a 10K overspend was acceptable.

Table 7: Risk Reduction Actions on Development Risks

Development Risk	Risk Reduct-ion	Cost (000's)	New Probability	New Impact (000's)
DR_1	DRR_1	1	1	5
DR_2	DRR_2	2	2	2
DR_2	DRR_3	2	1	1
DR_3	DRR_4	2	0	0
DR_4	DRR_5	2	0	0
DR_5	DRR_6	1	3	2
DR_5	DRR_7	1	0	0
DR_6	DRR_8	2	0	0
DR_7	DRR_9	1	1	6
DR_8	DRR_9	1	0	0
DR_9	DRR_9	1	1	3
DR_{10}	DRR_{10}	3	2	4
DR_{11}	DRR_{10}	3	1	2
DR_{12}	DRR_{10}	3	1	8

4.1. Simulation

Simulation was achieved by Monte Carlo Analysis with Latin Hypercube sampling. This is made possible due to the fact that we have a probability value and an impact value for each of the risks identified, allowing a discrete probability distribution to be generated. The probability value is assumed to be over the lifetime of the system. In the sample project an arbitrary 1000 iterations were used as a compromise between obtaining a good distribution and execution time. When applying Monte Carlo simulation to existing operational risks (Table 1 and 4), the probability value is that achieved after any risk reducing change has occurred if that change has been selected. Also operational risks introduced into the system (Table 5), will only apply if a candidate system change has been selected for implementation. Development risks (Table 6) are also represented by a simulated probability distribution based on the estimated probability and impact of each risk, if it occurs. These are also only taken into account if the associated change has been selected. Changes are selected using a genetic algorithm. This and the Monte Carlo analysis were performed using the RiskOptimizer software tool from Palisade [8].

Each system change (table 2) along with each risk reduction activity (table 3) is considered a candidate action. The genetic algorithm is used to select which of these are taken. The crossover routine used is that of uniform crossover as described in [2] and recommended in [9]. This involves randomly choosing items in each of 2 selected parents to use to create an offspring. The percentage in the first parent chosen is determined by the crossover rate. The decision to include a change or not is a binary one, so that crossover means taking some of the changes from one chromosome (generated solution), adding these to a new chromosome and then adding the changes from a different selected chromosome. Mutation is performed on the offspring by generating a random number between 0 and 1 for each item. If the number is less than or equal to the mutation rate then that variable is mutated. In our case the decision is whether or not a system change is included. In this case a mutation is effectively moving from exclusion to inclusion or vice versa. Following preliminary investigations a crossover rate of 0.5 was used and an auto-mutation routine employed that adjusted the mutation rate automatically when a population ceased to improve.

4.2. Sample Results

Table 8 provides details of the results produced using the data from the case study. The objective was to maximise the profit from the activities selected.

The results show a general exclusion of changes C_1, C_3 and C_6 from the enhancement type changes and of C13 from the operational risk reductions. This is probably a result of the mutual exclusion set μ (1). In the sample project, development risk reduction DRR_5 is included in 7 of the top ten solutions with occasional inclusion of DRR_6, DRR_7 and DRR_8 in the top ten. C_1, C_3 and C_6 did occur much further down in the rankings.

In practice, an analyst may take the top-ranked solution but also consider the third ranked solution which, in this case, has the same content but includes the development risk reductions of DRR_5 and DRR_6 with a loss in profit of 1K. This is as an acknowledgement of the uncertainty of the input data and that the analyst should be able to take account of their own judgement.

Note that the profit values also include any risks that were realised in the simulation. The operational risk reduction, C_{13} and several development risk reductions did not get selected in any of the top ten solutions, implying that they were not worth the effort within the limitations of the project.

Table 8: Top ten ranked solutions

Mean Profit		129.6	129.1	128.6	127.6	117.1	116.6	116.1	115.6	115.1	112.0
Min.		86.5	83.5	85.5	84.5	63.5	60.5	62.5	59.5	68.0	62.5
Max.		172.5	169.5	171.5	170.5	167.0	164.0	166.0	163.0	158.5	154.5
Rank		1	2	3	4	5	6	7	8	9	10
C_2	System Changes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_4		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_5		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_7		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_8		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_9		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_{10}		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_{11}	Op. Risk Red.	✓	✓	✓	✓	✓	✓	✓	✓	✓	
C_{12}		✓	✓	✓	✓	✓	✓	✓	✓		✓
C_{14}		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C_{15}		✓	✓	✓	✓					✓	✓
DRR_5	Dev. Risk Red.	✓		✓	✓	✓		✓		✓	✓
DRR_6				✓							
DRR_7								✓	✓		
DRR_8					✓						

5. Conclusions and Future Research

An approach for supporting the decision process in planning the evolutions of existing systems has been described. The approach is novel in 3 ways. (i) As well as considering corrective and enhancive work, operational risks in the current system are considered as identification sources for system change. (ii) Risks are modelled along with predicted costs using estimated probability and impact values and using Monte Carlo simulation. This is not novel in financial applications, but is regarding software evolution. (iii) A genetic algorithm has been used to optimise the system evolution plan for profit but maintaining a cost limit, taking account of inherent dependencies and ensuring development and operational risk exposure levels are kept below some limit. Overall, this decision support approach offers an improvement on the ad hoc approach currently prevalent, in taking account of risks as well as costs and benefits allows for a truer prediction of the likely costs of selecting certain changes, of accepting certain risks and mitigating others.

There remains some practical problems in collecting cost, benefit and risk data. One promising approach, from a previous industrial study [4] is the use of symbols to represent costs, benefits and probabilities and impact costs for risks. This approach involved choosing phrases such as “high” to represent the measure and mapping this to a preset value. This work was successful in capturing cost and risk data, requiring minimal expertise from the project planner, other than past experience. Incorporating this work further constitutes future work which will also involve further empirical studies of the approach, and the various trade-offs involved.

6. References

[1] Charette, R.N.; Adams, K.M.; White, M.B, Managing Risk in Software Maintenance, *IEEE Software*, Volume: 14 (3), 1997. pp. 43 – 50.

[2] De Jong, K A. *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*, Doctoral Thesis, Univ. Michigan, Ann Arbor 1975.

[3] Dorofee, A., Walker, J., Alberts, C.J., Higuera, R.P., Murphy, R.L. and Williams, R.C., *Continuous Risk Management Guidebook, Software Engineering Institute, Carnegie Mellon Univ*ersity, 1996.

[4] Greer, D., Bustard, D. and Sunazuka, T. Prioritisation of System Changes using Cost-Benefit and Risk Assessments, Fourth IEEE International Symposium on Requirements Engineering, pp 180-187, June, 1999.

[5] Greer, D. & Ruhe, G., Software Release Planning: An Evolutionary and Iterative Approach, J. Information and Software Technology, vol. 46, issue 4, pp 243-253, 2004.

[6] IEEE. 1993. *IEEE Standard for Software Maintenance*. IEEE Std 1219-1993. IEEE, New York, NY.

[7] Kemerer, C.F. & S. Slaughter, Determinants of Software Maintenance Profiles: An Empirical Investigation, *J. Software Maintenance*, vol. 9, 1997, pp. 235-251.

[8] Palisade Corporation, *Guide to RISKOptimizer: Release 1.0*, October, 2001, Palisade Corp.

[9] Spears, W.M.; De Jong, K.A., On the virtues of parameterized uniform crossover, Proc. International Conference on Genetic Algorithms, pp. 230-237, 1991.