Diversifying Software Architecture for Sustainability: A Value-based Perspective

Dalia Sobhy[1], Rami Bahsoon[1], Leandro Minku[2], and Rick Kazman[3]

[1] School of Computer Science, University of Birmingham, Birmingham, UK
dms446, r.bahsoon@cs.bham.ac.uk,
[2] School of Computer Science, University of Leicester, Leicester, UK
leandro.minku@leicester.ac.uk
[3] SEI/CMU, University of Hawaii, Honolulu, USA
kazman@hawaii.edu

Abstract. We use real options theory to evaluate the options of diversity in design by looking at the trade-offs between the cost and long-term value of different architectural strategies under uncertainty, given a set of scenarios of interest. As part of our approach, we extend one of the widely used architecture trade-offs analysis methods (Cost-Benefit Analysis Method) to incorporate diversification. We also use a case study to demonstrate how decision makers and architects can reason about sustainability using a diversified cost-value approach.

1 Introduction

Design Diversity is *"the approach in which the hardware and software elements that are to be used for multiple computations are not copies, but are independently designed to meet a system's requirements"* [2]. It is the generation of functionally equivalent versions of a software system, but implemented differently [2]. Design diversification has the potential to mitigate risks and improve the dependability in design for situations exhibiting uncertainty in operation, usage, etc. On the other hand, architecture sustainability is *"the architecture's capacity to endure different types of change through efficient maintenance and orderly evolution over its entire life cycle"* [1]. In this paper, we argue that we can link diversity and sustainability from a value-based perspective. The link can summarize the success of engineering and evolution decisions in meeting the current and future changes to users, system, and environment requirements. We are concerned with how to employ diversity in the architecture as a mechanism to better support future changes. This requires rethinking architecture design decisions by looking at their link to long-term value creation in enabling change and reducing their debt, etc. The focus is on how we can sustain the architecture, which requires treatment for not only short-term costs and benefits but also for long-term ones and their likely debts. As the valuation shall take into consideration uncertainty, we appeal to options thinking [7] to answer the above question. Our novel contribution is an architecture-centric method, which builds on Cost-Benefit Analysis Method (CBAM) [5] and options theory [7] to evaluate and reason about how architectural diversification decisions can be employed and their augmentation to long-term value creation. In particular, the approach uses real options analysis [7] to quantify the long-term contribution of these decisions to value and determine how that value can assist decision-makers and software architects in reasoning about sustainability in software. Our exploratory case analysis is based on provisional data gathered from the GridStix prototype, deployed at River Ribble in the North West England [4].

2 Background

CBAM: A Cost-Benefit Analysis Method that intends to develop an economic model of sofware and systems that helps a designer select amongst different architectural options at design-time [5]. CBAM extends ATAM with explicit focus on the costs and benefits of the architecture decisions in meeting scenarios related to quality attributes (QA) as illustrated in figure 1. Interested reader can refer to [5] for more details.
Real Options Analysis: We view architecting for sustainability through diversification as an option problem. Real options analysis is well-known paradigm used for strategic decision-making [7]. It emphasizes the value-generating power of flexibility under uncertainty. An option is the right, but not the obligation, to make an investment decision in accordance to given circumstances for a particular duration into the future, ending with an expiration date [7]. Real options are typically used for real assets (non-financial), such as a property or a new product design. We used call options, which give the right to buy an uncertain future valued asset for the strike price by a specified date. In this paper, we consider different architectural strategies and different options, and use the Binomial option pricing model [7] to value real options. The choice of this model gives the architect the freedom to estimate the up and down in the value over time, backed up by their experience.
GridStix: We present a case study based on the GridStix prototype, a grid-based technique to support flood prediction by implementing embedded sensors with diverse networking technologies [4]. The water depth and flow rate of the river are continuously observed using sensors located along the river. Data are collected in real-time and dispatched over GPRS to a prediction model [4] for flood anticipation. GridStix has a highly dynamic environment, and is influenced by numerous QAs and different architectural components [4]. Our evaluation, shown in section 5, is performed using hypothetical data, aiming to measure the long-term impact of implementing a diversified vs non-diversified design decisions on system QAs, cost, and value.

3 Architecture Diversification as a Real Options Problem

Diversified software architecture is composed of architectural strategies (ASs). It can meet some quality goals of interest and trade-offs by implementing a set of diversified ASs. At run-time, switching between diversified AS is allowed. Suppose

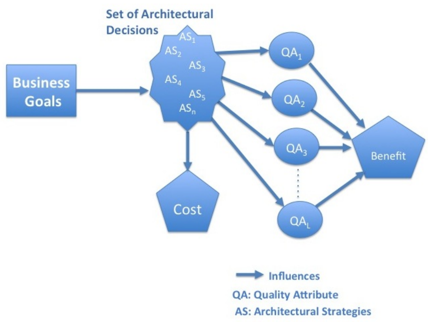


Fig. 1: Steps of Classical CBAM [5]

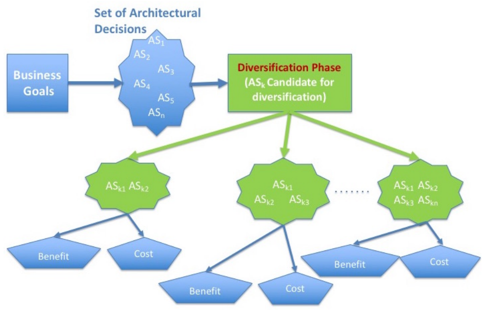


Fig. 2: Proposed Approach

that k denotes a particular capability, including connectivity, routing technology, data management, etc, as depicted in GridStix. AS_{ka} indicates the software architectural component a implementing capability k. Some of the following ASs are envisioned as a way to implement diversification in GridStix: AS_{11}, AS_{12}, AS_{13} are connect node with gateway via *Wifi*, *Bluetooth (BT)*, and *GPRS* respectively; AS_{21}, AS_{22} are search for the best path between gateway and node using *Fewest Hop (FH)*, *Shortest Path (SP)* routing algorithm, respectively.

Inspired by options theory [7], we consider each different possible diversified architecture as an option. Therefore, we refer to them as Diversified Architecture Options (DAOs). An example of DAO would be $DAO_1 = (AS_{11}, AS_{12}, AS_{21})$, meaning that the system can switch between AS_{11} and AS_{12} at run-time. Another example would be $DAO_2 = (AS_{11}, AS_{12}, AS_{13}, AS_{21}, AS_{22})$, meaning that the system can switch between AS_{11}, AS_{12} and AS_{13}, and between AS_{21} and AS_{22} at run-time.

The value of these options is long-term and can cross-cut many dimensions. In particular, the valuation of the options can be performed in accordance to sustainability dimensions, which can be technical, individual, economics, environment, and social [3]. In this paper, we attempt to link technical decision to cost and long-term value. When evolving an existing system, the current implementation of the system has a direct ramification on the selection of a DAO. It could provide an intuitive indication about whether the current system architecture needs to grow, alter, defer, etc. To exemplify, if the current system architecture has low long-term value, hence it is obvious that another DAO should be employed instead.

The goal of our approach is to help the architect to choose a DAO that provides a good trade-off between cost and long term value, given some quality goals. This is done by evaluating a portfolio of DAOs.

4 The approach

The proposed approach is a CBAM-based method for evaluating diversified architectural options (DAO) with real options theory, as illustrated in figure 2.

Step 1: Choosing the business goals, Scenarios and DAOs Our method focuses on QAs and their responses with respect to scenarios of interest that are related to sustainability. DAOs are the architectural options that deal with these scenarios. In our approach, DAOs are represented as a portfolio of options. Exercising each DAO can be formulated as call option [7], with an exercise price and uncertain value. We aim to provide a good trade-off between the benefit and cost of applying diversified options on system's QA over time, given the following: *1- A set of diversified architectural options* $\{DAO_1, DAO_2, DAO_n\}$*, where each DAO is composed of integrated architectural strategies among candidate diversified ones* $\{AS_1, AS_2, AS_m\}$*. 2- One or more ASs are selected as candidates for diversification* AS_k*, as shown in figure 2. 3-The diversified ASs are denoted by* AS_{ka}*, where* $0 <= k <= x, 0 <= a <= y$*. 4- Each DAO comes with a cost* $Cost_{DAO_i}(t)$ *and a benefit* $Benefit_{DAO_i}(t)$*, which may vary over time.*

Table 1: Approach Notations

Variable	Description	Formulation/Application on GridStix
DAO_i	Diversified Architectural Options	DAOs as a portfolio of call options
QA	Quality Attribute	Performance (Perf), Reliability (Rel), Availability (Ava), Security (Sec), Scalability (Sca), & Energy Efficiency (Ene)
$QAWeight_j$	The relative importance of QAs	should satisfy $\sum_j QAWeight_j = 100$, Perf(20), Rel(30), Ava(20), Sec(5), Sca(5), & Ene(15)
$ContribScore_j$	Impact of each DAO on QAs	DAO_1: Perf(1),Rel(1), Ava(0.8), Sec(0.5), Sca(0.7), Ene(-0.4),Cost(60)
$Benefit_{DAO_i}$	Benefit of a DAO	$Benefit_{DAO_i} = \sum_j QAWeight_j * ContribScore_{i,j}$
$Cost(DAO_i)$	Switching, Deployment, Configuration,& maintenance costs	$\forall i : Cost(DAO_i) \leqslant Budget$
S_{DAO_i}	System value	$S_{DAO_i} = V_s + Benefit_{DAO_i}$
$S_{DAO_i}(t)$	System value over time t	$S_{DAO_i}(t) = V_s + Benefit_{DAO_i}(t)$
u, d	system value (corresponding to stock price) benefiting/being hurt from DAO i.e. value rise	should satisfy $d < 1 + r < u$ [6]
r	Risk-free Interest Rate	0.5%
f_u	The likely rise of payoff from implementing a DAO	$f_u = max(0, uS_{DAO_i}(t) - Cost(DAO_i))$
f_d	The likely fall of payoff from implementing a DAO	$f_d = max(0, dS_{DAO_i}(t) - Cost(DAO_i))$
p	Risk-adjusted probability	$p = \frac{1+r-d}{u-d}$
f	Option Price	$f = \frac{pf_u+(1-p)f_d}{1-r}$

Among the business goals, which we consider to illustrate our approach are the accuracy of flood anticipation and reasonable warning time prior to the flood. In our method, we mainly test and evaluate the application of diversity versus no diversity. Therefore, $non-diversified-option = Wifi + FH$, $DAO_1 = Wifi + BT + FH$, and the following scenario *Messages transmission between any given sensor node and gateway should arrive in* $\leq$ *30ms (addressing the performance QA)* are employed for evaluation. We set 60% target for improvement of average network latency backed up by [4].

Step 2: Assessing the relative importance of QAs (Elicit QAWeight$_\mathbf{j}$) The architect assigned a weight to the QA according to equation in table 1.

Step 3: Quantifying the benefits of the DAOs (Elicit ContribScore$_\mathbf{j}$) The impact of *Non-diversified option* and DAO_i on the QAs are elicited from the stakeholders with respect to $Benefit_{DAO_i}$ equation in table 1.

Step 4: Quantifying the costs of DAOs and Incorporating Scheduling implications Classical CBAM uses the common measures for determining the costs, which involves the implementation costs only. Unlike CBAM, our approach embraces the switching costs between decisions, which is equivalent to the primary payment required for purchasing a stock option. This is in addition to the costs of deploying DAOs, configuration costs, and maintenance costs, similarly to the exercise price, denoted by *Cost* (DAO_i). It is essential to note that CBAM implements the ASs with high benefit and low cost [5]. On the other hand, we believe that some ASs could provide high cost with low benefit initially or high cost with high benefit, but a much higher benefit in the long-term that outweighs the cost. The long-term benefit is the key factor for ASs evaluation.

Step 5: Calculate the Return of each DAO for the scenarios We used binomial option pricing calculation [7] and steps inspired by [6]. Binomial option pricing model is a constructive aid aiming to show the suitable time slot for exercising an option i.e the cost-benefit of diversified options over time. For each step of the binomial tree, the up and down node values are important in determining the system value rise and fall, which is ultimately used to calculate the option price. Our method aims to determine the impact of applying each DAO (i.e utility) on the system QAs, which is computed at every time slot t, where t=l indicates that the time equals to l unit time of interest i.e months in GridStix. For example, currently, the approximate number of deployed gridstix nodes is 14 [4]. It is likely that adding extra nodes may improve the system's safety due to the presence of backup nodes and providing wider network coverage. This in turn promotes the accuracy of flood prediction, satisfying our main business driver, thus sustaining the GridStix software. Figure 3 envisages the enhanced utility gained with/without diversification versus reporting latency in accordance to offering up to 20 nodes, based on the graphs in [4]. The following steps are necessary for valuation of options using the binomial option pricing model.

1. ***Calculate the system value after factoring diversification into the decisions:*** As a start, S_{DAO} is evaluated with respect to the initial system value denoted by V_s and resultant benefit of deploying DAO as shown in

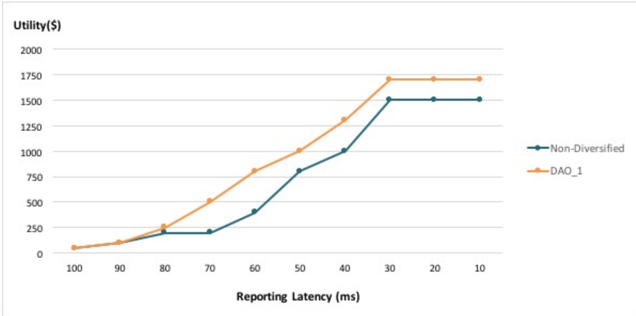


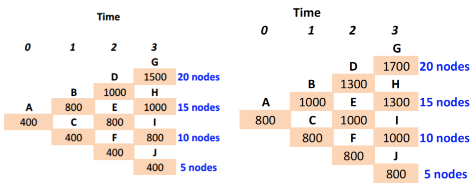
Fig. 3: Enhanced Utility versus Reporting Latency in case of implementing additional nodes for *non-diversified and diversified* decisions

table 1. Also $S_{\text{DAO}}(t)$ is the system value after implementing a particular DAO causing either incremental improvement or degradation at time t, which is equivalent to the uncertain stock price when modeling an American call option.

2. ***Calculate the likely rise and fall of payoff with DAOs*** f_u and f_d are computed using equations depicted in table 1.
3. ***Calculate the option price of exercising a DAO*** This step reveals at what time t, it is favorable to take the decision i.e. exercise an option using f as seen in table 1. It also illustrates the long-term performance of a system, which in turn aids in promoting sustainability.

5 Preliminary Evaluation

Without Diversification Outcome: A preliminary analysis of the method without diversifying ASs is necessary. The architecture comprising *Wifi* and *FH* was evaluated. The utility values for the implementation of the latter architecture is depicted in figure 4 along with utilities of other DAOs, which are elicited from stakeholders. Decision makers can vary the base value at cell A (guided by the chart in figure 4a) to perform what-if analysis. In this example, possible values range from \$400 to \$1500. The likely value of each architecture is different. The valuation of non-diversified option over varying time slots for uncertainty of implementing additional nodes is clearly shown in figure (5). In this example, V_s is \$1750. For detailed analysis, consider cell D for the evaluation of two-unit time as presented in figure 5, which is the upper cell value: $S_{non-div}(2) = V_s + Benefit_{non-div}(2) = 1750 + 1000 = \$2750$. The lower cell value is computed as follows: $f_{non-div}(2) = max(0, S_{uu} - Cost_{non-div}) = max(0, 2750 - 1250) = \$1500$. The option price formula f of non-diversified is:$f_{non-div} = f_{DAO_{non-div_1}} + f_{non-div_2} + f_{non-div_3} = 905.47 + 910.79 + 915.60 = \$2732.22$.
Diversification Outcome: DAO_1 is employed for method evaluation. The predicted utility values for the implementation of DAO_1 are revealed in figure 4b,



(a) Non-Diversified (b) DAO_1

Fig. 4: Anticipated Values for the utility of non-diversified and DAO_1

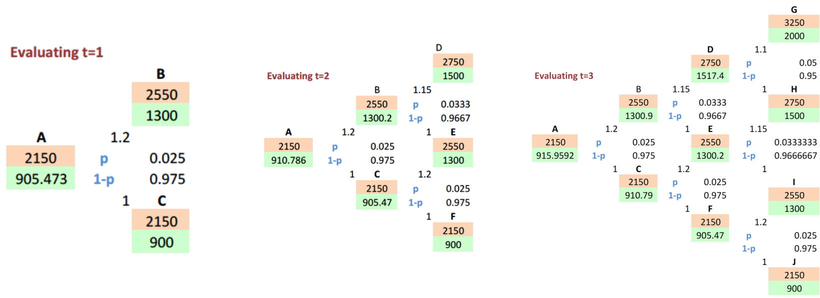


Fig. 5: Valuation of *non-diversified* option staged over 3 time periods

which is elicited from stakeholders. By applying the same logic used to calculate the option value for non-diversified decision, the valuation of DAO_1 over varying time slots for uncertainty of implementing additional nodes is shown in figure (6), where the orange cells represent $f_{DAO_i}(t)$ and green cells denote the $S_{DAO_i}(t)$. For detailed analysis, Consider cell D for the evaluation of two-unit time as presented in figure 6, which is the upper cell value: $S_{DAO_1}(2) = V_s + Benefit_{DAO_1}(2) = 1750 + 1300 = \$3050$. The lower cell value is computed as follows: $f_{DAO_4}(2) = max(0, S_{uu} - Cost(DAO_4)) = max(0, 3050 - 1500) = \$1550$. Therefore, the option price formula f of DAO_1 is: $f_{DAO_1} = f_{DAO_{1.1}} + f_{DAO_{1.2}} + f_{DAO_{1.3}} = 1049.8 + 1049.60 + 1049.56 = \$3148.91$.

Summary of Evaluation: From the above, the value of non-diversified is \$2750 and the value of DAO_1 is \$3150. The costs are \$2732.22 and \$3148.91, respectively. Although DAO_1 has higher cost than non-diversified option, yet it has higher long-term benefit. This proves that implementing high cost options would provide higher long-term benefit i.e high option value.

6 Conclusion

We have described an approach, which makes a novel extension of CBAM. The approach reasons about diversification in software architecture design decisions using real options. The fundamental premise is that diversification embeds flex-

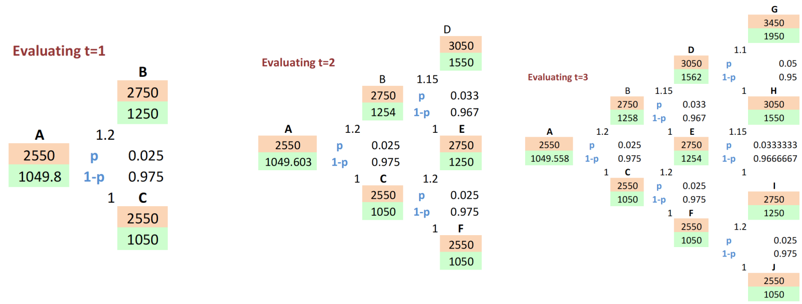


Fig. 6: Valuation of diversified option (DAO_1) staged over 3 time periods

ibility in an architecture. This flexibility can have value under uncertainty and can be reasoned using Real Options. In particular, the approach can be used by the architect and the decision maker to apprise the value of architecting for sustainability via diversification based on binomial trees. For instance, the method can be used to inform whether an architecture decision needs to be diversified and what the trade-offs between cost and long term value resulting from diversification are. This trade-off can be used to reflect on sustainability. Our case study illustrates that the method can provide systematic assessment for the interlink between sustainability and diversity using value-based reasoning. In the future, we plan to evaluate our model at run-time using machine learning techniques as well as apply it on several case studies.

References

1. Avgeriou, P., Stal, M., Hilliard, R.: Architecture sustainability [guest editors' introduction]. Software, IEEE 30(6), 40–44 (2013)
2. Avizienis, A., Kelly, J.P.J.: Fault tolerance by design diversity: Concepts and experiments. Computer 17(8), 67–80 (Aug 1984), `http://dx.doi.org/10.1109/MC.1984.1659219`
3. Becker, C., Chitchyan, R., Duboc, L., Easterbrook, S., Mahaux, M., Penzenstadler, B., Rodriguez-Navas, G., Salinesi, C., Seyff, N., Venters, C., et al.: The karlskrona manifesto for sustainability design. arXiv preprint arXiv:1410.6968 (2014)
4. Grace, P., Hughes, D., Porter, B., Blair, G.S., Coulson, G., Taiani, F.: Experiences with open overlays: a middleware approach to network heterogeneity. ACM SIGOPS Operating Systems Review 42(4), 123–136 (2008)
5. Kazman, R., Asundi, J., Klein, M.: Quantifying the costs and benefits of architectural decisions. In: Proceedings of the 23rd international conference on Software engineering. pp. 297–306. IEEE Computer Society (2001)
6. Ozkaya, I., Kazman, R., Klein, M.: Quality-attribute based economic valuation of architectural patterns. In: Economics of Software and Computation, 2007. ESC'07. First International Workshop on the. pp. 5–5. IEEE (2007)
7. Trigeorgis, L.: Real options: Managerial flexibility and strategy in resource allocation. MIT press (1996)