Prioritization of EA Debts Facilitating Portfolio Theory

Yoon Chow Yeong
Universiti Teknologi Petronas
Perak Darul Ridzuan, Malaysia
yeongyoonchow@gmail.com

Simon Hacks
Division of Network and Systems Engineering
KTH Royal Institute of Technology
Stockholm, Sweden
shacks@kth.se

Horst Lichter
Research Group Software Construction
RWTH Aachen University
Aachen, Germany
lichter@swc.rwth-aachen.de

***Abstract*—Implementing an enterprise architecture (EA) project might not always be a success due to uncertainty and unavailability of resources. Hitherto, we have proposed a new metaphor –Enterprise Architecture Debt (EAD)–, which makes bad habits within EAs explicit. We anticipate that the accumulation of EAD will negatively influence EA quality, also expose the business into risk.**
Recognizing the importance of business-IT alignment in enterprise architecture context, this paper proposes an application of portfolio-based thinking and utility theory for EAD prioritization. For proof-of-concept purpose, we develop synthetic data using coarse-grained estimates to demonstrate the application of the proposed portfolio-based approach which helps to determine the optimum selection of EAD to be resolved. The results show that our approach can help EA practitioners and management to reason their EA investment decisions based on the EAD concept, with adjustable enterprises risk tolerance level.

***Index Terms*—Enterprise Architecture Management, Enterprise Architecture Debt (EAD), Portfolio Theory, EA Portfolio Optimization, Utility Theory**

I. Introduction

Technical debt is a metaphor that had been introduced by Cunningham [1]. In the software development industry, technical debt is regarded as a critical issue in terms of the negative consequences such as increased software development cost, low product quality, decreased maintainability, and slowed progress to the long-term success of developing software [2]. Technical debt describes the delayed technical development activities for getting short-term payoffs such as a timely release of a specific software [3]. Seaman et al. [4] described technical debt as a situation in which software developers accept compromises in one dimension to meet an urgent demand in another dimension and eventually resulted in higher costs to restore the health of the system in future.

Furthermore, technical debt is explained as the effect of immature software artifacts, which requires extra effort on software maintenance in the future [5]. The concept of technical debt reflects technical compromises that provide short-term benefit by sacrificing the long-term health of a software system [6]. In view of the original idea of technical debt that focused on the code level in software implementation, the concept had been extended to software architecture, documentation, requirements, and testing [7]. While the technical debt metaphor has further extended to include database design debt, which describes the immature database design decisions [8], the context of technical debt is still limited to the technological aspects.

Over the years, technical debt becomes increasingly important when organizations invest huge amounts of money in IT to stay competitive, effective, and efficient. However, it is vital to align IT and business in order to realize the full benefits and potentials of those IT investments [9]. From there, the concept of Enterprise Architecture (EA) has evolved as a method to facilitate the alignment of IT systems and business strategies within dynamic and complex organizations [10]. Consequently, the huge interest in EA resulted in vast scientific contributions that address a broad thematic spectrum [11], including EA frameworks, EA management, and EA tools. However, there is a lack of insight into the application of the debt concept to include not only the technological aspects addressed by technical debt, but also the business aspects. Adapting the concept of technical debt in the EA domain, hitherto we have proposed a new metaphor "Enterprise Architecture debt (EAD)" to provide a holistic view [12].

In the real world, debt is not necessarily a negative thing to incur, same goes to EA debt to be held in an enterprise. The danger of debt comes into place when there is no proper debt management approach to prioritize, which debt should be repaid as soon as possible. We predict that managing EA debt will be one of the critical success factors of EA implementation and, thus, there is tremendous need to allocate resource effectively to maintain the current level of profitability by properly managing EA debts that exist in an enterprise.

Numerous studies have been dealing with the approaches to prioritize technical debt in the domain of software engineering [3]–[5], [8], [13]–[16], and yet these studies do not address the business aspects as a whole EA. To fill the research gap, this study aims to extend the application of portfolio theory into the concept of EA debt. This will be achieved by focusing on the following research questions:

(*RQ1*) *How can a given set of EA debt items be prioritized based on a portfolio approach?*

The following list of research sub-questions are emerged from the main research question which is mentioned as above:

(*RQ1.1*) *What attributes of EA debt should be contained in a portfolio-based prioritization model?*

(*RQ1.2*) *What are the process steps required to prioritize EA debt items based on a portfolio thinking?*

This study proposes a portfolio-based approach to prioritize EA debt that exists in EA implementation by incorporating the portfolio thinking and utility theory into EA. This proposed approach contributes to the theoretical body of knowledge by providing a fundamental understanding on how EA debt items can be conceptualized and measured for decision-making. It is strongly believed that this approach can measure, manage, and prioritize debt on an enterprise-wide level, which can be valuable to EA stakeholders by avoiding massive interests on EA debt. In light of the novel introduction of the EA debt metaphor, it is foreseen that EA debt decision-making would be a worthwhile subject for future research in the EA field.

The rest of this paper is structured as follows: First, we introduce the facilitated key concepts of modern portfolio theory and utility theory. Second, we present in Section III how we apply the concepts of portfolio theory and utility theory (Section III-A), and depict a process, which guides the prioritization (Section III-B). Next, we demonstrate our approach on a fictitious case study in Section IV and present related work (Section V). Last, we conclude our work in Section VI.

II. Key Concepts

A. Modern Portfolio Theory

In the finance domain, Modern Portfolio Theory (MPT) was originally developed by Markowitz [17]. The goal of this theory is to develop an approach to determine an efficient portfolio with the maximum return at a given level of risk or the minimum risk at a given level of return. Based on this, decisions can be made of which types and amounts of financial assets in a portfolio should be invested or divested [17], [18]. The investments can be stocks, bonds, or other financial products that are characterized by a return at a certain level of risk. The development of MPT was based on the rule that investors should consider expected or anticipated return as a desirable thing, whereas variance of return as an undesirable thing. In MPT, a portfolio is a weighted combination of assets in which each asset's return and variance of the return are used to measure the portfolio performance [17].

The fundamental concept behind MPT emphasizes the importance of evaluating the relationship between price changes in each asset and price changes in every other asset in the portfolio [19]. Each individual financial asset generates different level of return and risk and, thus, the introduction of MPT seeks to minimize the total variance of the investment portfolio's return through the concept of diversification. The diversification allows investors to combine different assets whose returns are not perfectly positively correlated. By wisely deciding on the proportions of various financial assets, the advantage of diversification can be achieved through the portfolio return maximization for a given level of portfolio risk, or the portfolio risk minimization for a given level of portfolio return.

The expected return of a portfolio is expressed by the following equation [17]:

$$E = \sum_{i=1}^{N} w_i \mu_i \tag{1}$$

where E is the portfolio's return, w_i is the weight of asset i in the portfolio, the sum of all weights w has to be 1, and μ_i is the expected return of asset i. On the other hand, the portfolio variance of return is calculated as follows [17]:

$$V(E) = \sum_{i=1}^{N} w_i^2 \delta_i^2 + 2 \sum_{i<j}^{N} w_i w_j \rho_{ij} \tag{2}$$

where $V(E)$ is the portfolio's return variance, δ_i is the return variance of individual asset i, and ρ_{ij} is the covariance between the assets i and j. The portfolio's standard deviation, δ_P can then be computed as follows:

$$\delta_P = \sqrt{V(E)} \tag{3}$$

Often, MPT relies on historical variance of financial assets' returns to measure the risk. Unfortunately, projects that involve non-financial assets commonly do not have well-defined historical variance for absolute objective measurement [19]. However, Omisore et al. [19] asserted that this does not eliminate the possibility of applying MPT to non-financial assets because the concept is transferable to a wide range of investments as long as the "risk" is expressed in terms of uncertainty about expectations and possible losses on forecasts. Therefore, in EA debt context, we express risk in term of "chance of interest growth", which brings the risk of an increased amount of required effort to resolve an EA debt item in future phase as well as the negative impacts on the EA value.

B. Utility Function and Risk Aversion

In general, most investors require a greater return as compensation for taking a greater risk [19]. Nevertheless, investors differ in their level of risk tolerance, which means that they are risk averse to varying degrees and eventually leads to different utility functions [20]. The concept of utility function provides a way to select the optimal portfolio that yields the best trade-offs of return and risk, and gives the most satisfaction (utility) to the investors, taking their risk tolerance level into consideration [21]. This can be applied in the enterprise context where each profit-seeking enterprise differs in the amount of risk it is willing to accept at a given level of return.

To address the differences in enterprises' risk tolerance level, our approach proposes to prioritize EA debt portfolio for repayment based on the portfolio theory along with the principle of expected utility maximization. The utility maximization principle states that a rational investor acts to choose an investment that maximizes the expected utility of wealth among a set of feasible investment alternatives [20]. The similar concept can be applied in the context of EA debt in which an enterprise should act to invest in paying off the EA debt portfolio that maximizes the expected utility of resources among a set of existing EA debt portfolios.

Since risk aversion is not an objectively measurably quantity that aims for absolute measure, there is no unique utility function, which comes into place. Carlsson et al. [21] reported that one of the commonly employed utility function is:

$$U(P) = R_P - 0.005 * A * \delta_P^2, \quad (4)$$

where R_P is the expected return of a portfolio, A is an index of the investor's risk aversion coefficient (a higher index indicates a higher level of risk averseness), and δ_P^2 is the variance of the portfolio's expected rate of return, which is the square of standard deviation, δ_P, a measure of portfolio risk. The risk aversion coefficient is meant to be positive for all risk-averse investors whereas a negative index indicates a risk-loving investor [18].

The factor of 0.005 in Equation (4) is a scaling convention and normalizing factor that allows us to express the R_P and δ_P^2 as a percentage value instead of decimals. Adhering the positive affine transformation property of a utility function, we are allowed to scale a utility function by translating it with the addition and/or subtraction of any constant [20]. To further scale down the size of variance for easy interpretation in our study, we reduce the factor of 0.005 in Equation (4) to 0.001.

In this work, we apply the altered Equation (4) to plot risk-indifference curves (also known as utility curves) which allows us to select the attainable and optimum debt portfolio by combining the curves with the risk-return trade-off plots. Having to say that the single point where one of the curves intersects the efficient frontier is the debt portfolio that provides the best combination of risk-return for the risk level that is acceptable for the organization.

III. Applying Modern Portfolio Theory to Enterprise Architecture debt

In the field of Information Systems (IS), it is common to apply theories that originates from a diverse set of disciplines such as psychology, sociology, economics, finance, and computer science for problem-solving at the intersection of people, information technology, and organizations [22]. Based on the definition of EA debt presented by Hacks et al. [12], this work suggests the application of portfolio thinking into EA debt context with the aim of expanding the visibility and understanding of the newly introduced metaphor.

Technically, EA is responsible for translating the organizations strategy into projects that result in the achievement of a target state of the enterprise [10]. The gap between the target (to-be) architecture and the current (as-is) architecture is to be filled by identifying and implementing projects, programs, or initiatives. However, the required resources to achieve the goal of a project, program, or initiative are limited in terms of budget, time and performance specifications [23].

With inadequate resources and other forms of constraints, there are situations where we need to compromise certain principles, goals etc. in the first EA life cycle phase. Any omission of business and IT aspects inevitably leads to an incomplete view of the EA concerned and may result in an EA debt. As described by TOGAF [24], the Architecture Development Cycle (ADM) is a method to develop an EA in a continuous and iterative manner. From there, we anticipate that the existing EA debt somehow needs to be repaid in the future EA life cycle phase and, thus, EA practitioners, IT representatives, and management are accountable to make decisions on which EA debt item needs to be repaid first in order to avoid higher future cost.

It is expected that the accumulation of EA debt in an EA project will significantly affect the quality of an EA such as its maintainability and agility in responding to the rapidly changing business environments. Oppositely, if the EA debt is managed effectively, it is expected to increase the EA value. In other words, EA debt is analogous to a financial asset that generates return at a certain amount of risk. We expect that EA debt prioritization helps to effectively pay off the EA debt and, in turn, the EA can be adapted towards the new business requirements.

In view of the similarity between financial investment and incurring EA debt, we realize a potential of mapping the concept of financial portfolio management to the EA debt context.

A. *How to Measure EA debts*

An organization, which intends to implement EA, consists of numerous EA projects for enterprise transformation, e.g. adding new business processes or retiring applications. This study reasons that EA debt items inevitably exist in each project. EA debt items could be the failure of removing outdated elements in diagrams, a missing implementation standard, undefined business role definitions, outdated technological structure, etc. EA debt prioritization is a decision-making process that involves determining, which EA debt portfolio is optimal to pay off in the current phase. By examining the EA debt items across business and IT layers, an enterprise can gain a better understanding of EA debt embedded along the process of EA implementation. Having EA debt properly assessed and being paid off in line with the long-term business mission and goals, we can ensure that the resources are utilized efficiently. In short, EA debt measurement urges a new way of thinking of technical debt as an integrated part of the organization's business aspects.

In order to apply financial portfolio theory to EA debt, we need to quantify EA debt for measurement. Therefore, we derive a set of operational definitions in line with the financial definitions in portfolio theory that conform to the common assumptions of existing technical debt studies [5], [8]. Based on existing technical debt literature [5], [8], [14], [15], [25], each EA debt item has its associated principal estimate, interest estimate, and expected return. Despite the unit used for technical debt measurement in dollars, hours,

TABLE I
LIST OF MEASUREMENT ATTRIBUTES

Measurement attribute	Operational definition
Principal (X)	The number of hours required to resolve the EA debt in the current EA phase.
Interest amount (IA)	The extra hours that will be required in the future phase if the EA debt is not resolved in the current phase.
Interest variance (δ)	The likelihood that the interest amount will increase at the point of repayment in future phase.
Expected return (R)	Net benefit of resolving the EA debt in the current phase and hold the debt to the future phase.
Risk aversion coefficient (A)	The degree of enterprise's risk averseness.

people, or work units, these figures are easy to interpret and to handle, because they serve as a common language, allowing trend monitoring as well as historical data comparison [16]. This study opts to use working hours as measurement unit. Each measurement attribute is summarized in Table I.

Referring to Table I, we give an instance to provide an insight into how each measurement unit can be represented. Once an EA debt item is incurred in an EA project, a certain amount of working time is required to resolve the EA debt, which is denoted by principal (X). For instance, due to an enterprise architect's careless examination, one of the outdated elements in a use case diagram of System A was not removed. This EA debt item incurs an debt which requires a principal of 20 min to update the use case diagram immediately, at time 0 (t_0). However, if this particular debt item is held to a future phase (t_n), this will cause faultiness in clarifying system requirements being developed and eventually affects progress of system development.

As such, this EA debt item carries an interest amount (IA) of 16 min, which is the extra hours required in the future to identify and correct the faultiness, which already brings negative impacts. Unfortunately, this interest amount is uncertain in such a way that it is assumed to fluctuate over time depending on the scope, complexity, and impact of the components that the EA debt item is associated with at the time of repayment. In this case, if the EA debt item is held until time 3, t_3, the interest amount, is very likely to increase from $IA_{t_1} = 16$ to $IA_{t_4} = 96$ min, because the EA debt item is now not only bringing negative impacts to the planning phase, but also the development phase. This uncertainty of interest growth rate in the future represents the risk level of an EA debt item, because the EA debt item with high interest growth rate accumulate interest faster, which brings a higher future cost, which can be expressed as interest standard deviation (δ_d). If the interest growth rate of a particular EA debt item is not likely to grow or its growing rate is much lower than other EA debt items, this indicates that the debt payment can be deferred in a way that it carries lower risk.

On the other hand, the expected return of an EA debt item at time t can be understood as the number of working hours that can be saved by paying the debt at t_0. The expected return is calculated using the following equation:

$$R_d^t = (X(1 + \frac{IA}{X})^t) - X, \tag{5}$$

where R_d^t is the individual EA debt item's expected return at time t, X is the principal, and IA is the interest amount of the EA debt item.

To fit in the MPT model, we need to determine the "weight" of each EA debt item (w_d) and "correlations with other EA debt items" for each of the identified EA debt items. We assume that an EA debt portfolio contains all EA debt items in equal proportions, in a way that, $\sum_{w \in W} w_{d_i} = 1$. On the other hand, we adapt the idea of Guo and Seaman [5] to use correlation coefficients to represent the correlation between two debt items, where COR_{ij} expresses the correlation between d_i and d_j. Since an EA debt portfolio is made up with multiple EA debt items across multiple architectural layers, determining the correlations between EA debts requires analysis of all EA entities embedded in EAM activities as well as interoperability between architecture entities and architecture domains. A reliable estimation of correlations could be done through dependency analysis [5]. For simplicity, we consider that the correlation coefficient would be either 1 (two debt items are related to each other) or 0 (two debt items are unrelated to each other). With the value of correlation coefficients, a co-variance matrix can be created by computing:

$$\rho_{ij} = \delta_{d_i} \delta_{d_j} COR_{ij}, \tag{6}$$

where ρ_{ij} is the co-variance between debt items i and j.

With all the measurement attributes required in the MPT model, the expected return, variance, and standard deviation of an EA debt portfolio can be computed using the equations (7), (9), and (10), respectively. The expected return of an EA debt portfolio at time t is the weighted sum of its EA debt items' expected returns:

$$R_P^t = \sum_{i=1}^{n} w_{d_i} R_{d_i}^t, \tag{7}$$

with one constraint as presented in the following equation:

$$\sum_{i=1}^{n} w_{d_i} = 1. \tag{8}$$

On the other hand, the variance of the portfolio's return, which indicates the probabilities that the set of EA debt items will return different levels of benefits, is expressed as:

$$\delta_P^2 = \sum_{i=1}^{n} w_{d_i}^2 \delta_{d_i}^2 + 2 \sum_{i<j}^{n} w_{d_i} w_{d_j} \rho_{ij} \tag{9}$$

The EA debt portfolio's standard deviation, δ_P can then be computed as follows:

$$\delta_P = \sqrt{\delta_P^2} \tag{10}$$

As in practice, it is difficult to estimate and accurately model the exact amount of consequences of an EA debt item. Initially, when an EA debt item associated with each layer is identified, the principal, interest amount, and interest growth rate is estimated subjectively according to the enterprise architects experience. These rough estimations can then be adjusted using historical data that was collected throughout the EA life cycle. The more accurate and detailed the data is, the more reliable the estimation. The following section demonstrates how the proposed approach can be applied to reason about prioritization decisions.

B. Application Process

To apply portfolio theory and utility function to prioritize EA debt items for debt repayment, we need to ensure that the considerations mentioned in Section III-A are taken into account in order to map the portfolio model to EA debt measurement. Following, we propose a series of steps to identify the optimal EA debt portfolio based on portfolio model (cf. Figure 1). On top of that, with the application of utility function, enterprise architects can reason and justify about EA debt repayment decisions based on the enterprises risk tolerance level. The basic steps of our proposed prioritization approach are stated as follows:

1) Identify a project involved in EA implementation to achieve the target (to-be) architecture and let P be its EA debt portfolio.
2) Identify the associated EA debt items, $d_i \mid \forall i \in \{1, 2, ..., n\}$. This step is important, as not only debt items in the IT domain, but also in the business domain are identified.
3) For each EA debt item, d_i, estimate the principal (X_{d_i}), interest amount (IA_{d_i}), interest growth rate/interest standard deviation (δ_{d_i}), weights (w_{d_i}) and the correlations with other debt items ($COR_{d_{i,j}}$).
4) For each EA debt item, d_i, determine the values of portfolio model, which are the expected return (R_{d_i}) and the covariance matrix (ρ_{ij}) using Equation (5) and Equation (6), respectively.
5) Run the portfolio model on the available data to determine the expected return (R_P), variance (δ_P^2) and standard deviation (δ_P) of the EA debt portfolio.
6) Repeat steps 1-5 for all EA projects.
7) Identify the efficient EA debt portfolios. The efficient debt portfolios are the ones that lie on the efficient frontier and give the best return-risk trade-off if the debt is repaid at the current EA phase.
8) Determine the enterprise's risk aversion coefficient. For simplicity, this study ranges risk aversion coefficients from 1.0 to 5.0, with the lower number representing higher tolerance to risk.
9) Apply the utility function (Equation (4)) to calculate the risk-indifference curves.
10) Identify and prioritize the optimum portfolio where the utility curve intersects at the efficient frontier.

IV. Case Study

For proof-of-concept purpose, we applied a synthetic case study and artificial data was generated accordingly based on the proposed steps described in Section III-B. Coarse-grained estimates of EA debt items' properties have been made and we acknowledge that it is sufficient for measuring the EA debt items for preliminary prioritization decision-making. Estimates that are more detailed can be made when more real-world information is available upon which to base the estimates.

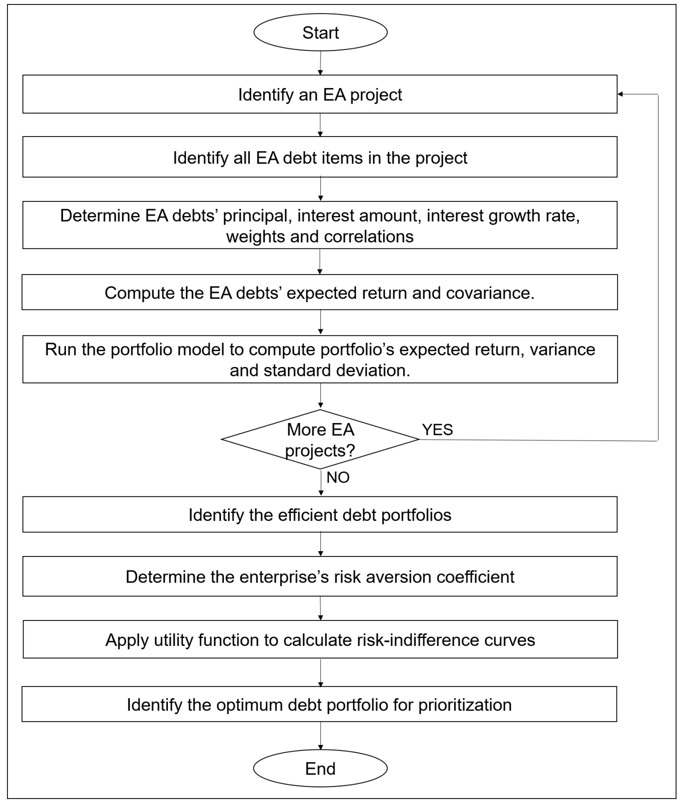

Fig. 1. Proposed Process Steps

We considered that a Company ABC can choose from five projects to support the enterprise transition from a current EA to a target EA in order to improve business-IT alignment. Along the EA implementation life cycle, various types of EA debt items incurred in each EA project.

To provide a better understanding of our proposed approach, the following data demonstrates the application of portfolio theory and utility function in the context of EA debt prioritization to show how an optimal EA debt portfolio can be identified and prioritized for decision-making.

Step 1: Identify an EA project: Project A (also known as EA debt portfolio A).

Step 2: Identify the associated EA debt items across the four architectural layers. See Table II.

Step 3: Estimate the principal, interest amount and interest growth rate of each EA debt item. See Table III.

Step 4: Compute the expected return and covariance matrix for each EA debt item. See Table IV.

Step 5: Run the portfolio model to compute the expected return, variance, and standard deviation of the EA debt portfolio. See Table V first row.

TABLE II
LIST OF EADS IN DEBT PORTFOLIO A

PROJECT A		
ID	**EAD domain**	**EAD description**
A1	Business Architecture	Fail to remove the outdated elements in a use case diagram
A2	Application Architecture	Fail to document interface descriptions
A3	Data Architecture	Lack of data model for an application
A4	Technology Architecture	Missing implementation standard

TABLE III
PRINCIPAL, INTEREST AMOUNT AND INTEREST GROWTH RATE ESTIMATES OF DEBT PORTFOLIO A

PROJECT A			
ID	**Principal**	**Interest amount**	**Interest growth rate**
A1	20 min	16 min	80 %
A2	16 min	11 min	70 %
A3	23 min	14 min	60 %
A4	8 min	4 min	50 %

Step 6: Identify other EA projects and repeat steps 2-5. Table V displays the computed portfolio's expected return and risk of five projects.

Step 7: Identify the efficient EA debt portfolios. As shown in Figure 2, debt portfolio A and B are inefficient portfolio because other portfolios can offer higher return at the similar level of risk or a lower risk at the similar level of return.

Step 8: Define the enterprise's risk aversion coefficient. Company ABC has a risk aversion coefficient value of 2.

Step 9: For visualization, calculate and plot the risk-indifference curves. See Figure 3 for exemplary curves for the utility values of 4,6,8, and 10.

Step 10: Identify the optimum portfolio for prioritization by solving Eq. 4 for every portfolio. The risk-return scatter plot in Figure 4 indicates that EA debt portfolio C is the optimum portfolio that provides best risk-return trade-offs and maximum satisfaction, as it (almost) based on the utility curve of 10.

Our synthetic case study shows that our approach is applicable in general. However, further research is necessary to enable enterprises to apply our approach in practice. Especially, steps 3 and 4 might be extremely challenging, due to missing experience in the field. To tackle step 3, we suggest collecting and documenting possible EA debt items and provide them as catalogs to the community. These catalogs can help to identify possible debt items and serve as a discussion basis to get a deeper understanding of the domain.

Step 4 requires the determination of the measurement attributes, which are needed to calculate the optimal portfolio. However, those attributes usually will be not obvious as for financial assets. Therefore, future research should elaborate on methods that enable practitioners to assess this attributes in an easy manner.

TABLE IV
PORTFOLIO MODEL VALUES OF DEBT PORTFOLIO A

PROJECT A							
ID	**Weight**	**Expected return**	**Standard deviation**	**Covariance matrix**			
				A1	A2	A3	A4
A1	0.25	16 min	0.8	0.64	0.56	0.48	0.4
A2	0.25	11 min	0.7	0.56	0.49	0.42	0.35
A3	0.25	14 min	0.6	0.48	0.42	0.36	0.3
A4	0.25	4 min	0.5	0.4	0.35	0.3	0.25

TABLE V
RISK AND EXPECTED RETURN OF 5 DEBT PORTFOLIOS

Project	Portfolio risk	Portfolio expected return
A	0.65	9.5
B	0.467	9.333
C	0.4	12.75
D	0.567	15.67
E	0.2	8.333

V. RELATED WORK

Despite the vast attention have been paid to technical debt, to our best knowledge, there is no existing approach to prioritize EA debt items as this metaphor is recently proposed by us [12]. Therefore, existing prioritization approaches have been studied in the context of technical debt.

Technical debt management (TDM) is composed of a sequence set of activities to prevent technical debt from being incurred or manage existing technical debt to maintain it under a desirable level [6]. TDM activities include TD identification, TD measurement, TD prioritization, TD prevention, TD monitoring, TD repayment, TD documentation, and TD communication [6]. Technical debt prioritization is considered as one of the TDM activities in which the identified technical debt items are ranked based on predefined rules to decide either immediate repayment or deferred repayment on the debt items [6]. Existing studies have discussed four decision approaches to deal with technical debt prioritization for complex decision-making: Cost-benefit analysis [3], Remediation cost analysis [16], Real Options [13] and Portfolio theory [4], [5]. Meanwhile, the systematic literature review on the financial aspect of managing technical debt conducted by Ampatzoglou et al. [25] concluded that the three most popular financial approaches are cost/benefit analysis, real-options analysis, and portfolio management.

Simple cost-benefit analysis: A simple cost-benefit approach was proposed to prioritize technical debt in terms of which classes should be re-factored first [3]. Each technical debt item consists of the estimations of three metrics: principal, interest probability, and interest amount. This approach prioritize code level debts based on the impact of the God classes on the software maintainability and correctness.

Remediation cost analysis: Moreover, the SQALE (Software Quality Assessment Based on Life-cycle Expectations) method with Sonar tool was proposed to analyze technical debt that associated with an application source code [16]. The authors proposed the synthesis of SQALE Quality and SQALE

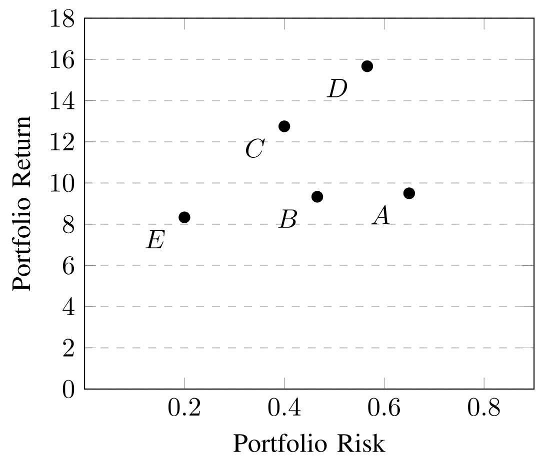

Fig. 2. Risk-return Trade-off of Five Debt Portfolios

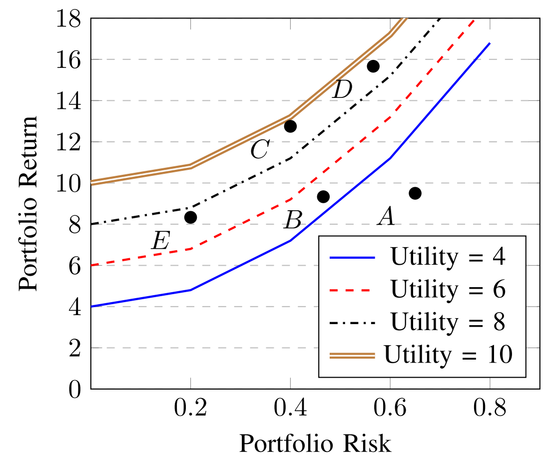

Fig. 4. The Optimal Debt Portfolio

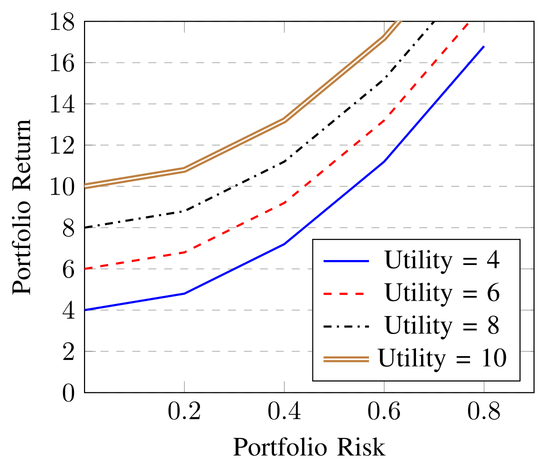

Fig. 3. Risk-indifference curves when risk aversion coefficient=2

Analysis model to measure technical debt in terms of the distance between the codes current quality state and its target state to indicate the quality of an application. On top of that, remediation index was used to represent the remediation cost of corrective actions required to resolve the non-compliance associated with each component of the applications software code.

Real-options approach: Another existing approach is incorporating real options thinking into the valuation of technical debt. Technically, the concept of a real option is about a right to make a future decision without any obligation depending on the way uncertainty is resolved. In other words, purchasing the real option is analogous to investing in paying off technical debt that facilitates future software changes. The real options theory was applied to effectively deal with unpredictable changes in system requirement engineering, time period, and development cost [13]. The proposed approach considers the risk associated with technical debt decisions to manage the value of an organization's strategic flexibility.

Portfolio theory: Furthermore, a portfolio approach was proposed to assist in decision-making in which technical debt items should be repaid and which one should be held for technical debt management [4], [5]. The measurement units embedded in portfolio model, such as expected return, return variance, and return standard deviation are mapped to the context of technical debt management. Each technical debt item was viewed as an asset and the application of portfolio mathematical formulation into a portfolio of debt items helps software developers to decide which technical debt items should be repaid first in order to minimize the future maintenance cost. Also, the portfolio theory was integrated into goal-obstacle method to specifically deal with requirements compliance debt [15].

Our approach differs from those works in two aspects. First, we broaden the scope of technical debt to the entire organization and propose a mapping of EA debt properties to portfolio theory properties. Second, the existing literature on applying portfolio theory to technical debt lacks an explicit description of its application, while we do.

Benchmarking and portfolio matrix: Instead of prioritizing technical debt in general, Plösch et al. [14] focus particularly on design debt, which is incurred due to the violations and non-conformance of design principles on source code level. The authors developed a tool called MUSE in which a portfolio matrix is used to prioritize the identified violations and to communicate design debt. In the proposed approach, a benchmarking-oriented measurement is applied to derive a quality index and categorize each design best practice into Q0- to Q5-area based on the number of identified design best practice violations.

Based on the analysis of existing literature, it is found that the limitation of the aforementioned approaches is that the relative importance of business impacts or operations are not taken into account. Therefore, the concept of linking EA debt to enterprise architecture and applying portfolio thinking is novel. While Stochel et al. [26] suggested to regard each distinct type of technical debt such as process debt as a debt portfolio, we suggest to map each EA project as a debt portfolio.

VI. Conclusion

Implementing EA is essential to enhance the business-IT alignment in a holistic manner. However, academia, software developers, and organizations have been focusing on technical debt, which deals with the quality issues on code, application, and system level. Considering the importance of EA in creating value to organizations, this work has explored a method to identify the optimal set of EA debt items, which should be repaid next. Therefore, we have elaborated on the necessary attributes of EA debts $(RQ1.1)$ and on the necessary process steps $(RQ1.2)$. To tackle $(RQ1.1)$, we have defined a mapping from the EA debt domain to the used terminology in portfolio optimization (see table I). This mapping is used as input for the process (see figure 1) to prioritize the EA debts that answers $(RQ1.2)$.

One of the limitations is that the portfolio-based EA debt model is developed based upon a high-level approach. This means any details, such as EA debt estimation tools and methods, are outside of the research scope. Therefore, estimation guidelines should be developed based on the professional experienced EA practitioners to provide the reference for estimating the debt principal and interest value of each assessed EA debt item.

In the meantime, coarse-grained estimations of EA debt measurement units have been made and we acknowledge that it is sufficient for prioritizing the EA debt items for preliminary decision-making. In real-world practice, EA practitioners are encouraged to substitute estimations based on historical measurements of extra costs required in EA debt repayment. More detailed planning can be made when more historical information is available upon which to facilitate the estimations.

Future research within this domain is two-fold. First, it is necessary to provide catalogs of EA debt items to enable practitioners to identify them in their EA. Those catalogs need to be validated and expanded. Further, effort should be invested to develop methods, which enable the practitioners to assess the measurement attributes that are needed to compute the optimal portfolio. Second, further means to prioritize technical debts need to be transferred to the domain of EA debts. Then, the different means need to be compared concerning their efficiency to determine the most efficient one.

References

[1] W. Cunningham, "The WyCash portfolio management system," *ACM SIGPLAN OOPS Messenger*, vol. 4, no. 2, pp. 29–30, 1993.

[2] E. Tom, A. Aurum, and R. Vidgen, "An exploration of technical debt," *Journal of Systems and Software*, vol. 86, no. 6, pp. 1498–1516, 2013.

[3] N. Zazworka, C. Seaman, and F. Shull, "Prioritizing Design Debt Investment Opportunities," *Proceeding of the 2nd working on Managing technical debt - MTD '11*, pp. 39–42, 2011.

[4] C. Seaman, Y. Guo, N. Zazworka, F. Shull, C. Izurieta, Y. Cai, and A. Vetrò, "Using technical debt data in decision making: Potential decision approaches," in *3rd International Workshop on Managing Technical Debt, MTD 2012 - Proceedings*, 2012, pp. 45–48.

[5] Y. Guo and C. Seaman, "A portfolio approach to technical debt management," in *Proceeding of the 2nd working on Managing technical debt - MTD '11*, 2011, p. 31. [Online]. Available: https://resources.sei.cmu.edu/asset_files/Presentation/2011_017_001_516999.pdfhttp://portal.acm.org/citation.cfm?doid=1985362.1985370

[6] Z. Li, P. Avgeriou, and P. Liang, "A systematic mapping study on technical debt and its management," *Journal of Systems and Software*, vol. 101, pp. 193–220, 2015.

[7] N. Brown, I. Ozkaya, R. Sangwan, C. Seaman, K. Sullivan, N. Zazworka, Y. Cai, Y. Guo, R. Kazman, M. Kim, P. Kruchten, E. Lim, A. MacCormack, and R. Nord, "Managing technical debt in software-reliant systems," *Proceedings of the FSE/SDP workshop on Future of software engineering research - FoSER '10*, no. January, p. 47, 2010.

[8] M. Albarak and R. Bahsoon, "Prioritizing technical debt in database normalization using portfolio theory and data quality metrics," in *Proceedings of the 2018 International Conference on Technical Debt - TechDebt '18*, 2018, pp. 31–40.

[9] D. H. Olsen, "Enterprise Architecture management challenges in the Norwegian health sector," *Procedia Computer Science*, vol. 121, pp. 637–645, 2017.

[10] J. A. Zachman, "A Framework for Information Systems Architecture," *IBM Systems Journal*, vol. 26, pp. 276–292, 1987.

[11] F. Gampfer, A. Jürgens, M. Müller, and R. Buchkremer, "Past, current and future trends in enterprise architectureA view beyond the horizon," *Computers in Industry*, vol. 100, pp. 70–84, 9 2018.

[12] S. Hacks, H. Höfert, J. Salentin, Y.-C. Yeong, and H. Lichter, "Towards the Definition of Enterprise Architecture Debts," *arXiv e-prints*, 2019. [Online]. Available: https://arxiv.org/pdf/1907.00677.pdf

[13] Z. S. H. Abad and G. Ruhe, "Using real options to manage Technical Debt in Requirements Engineering," *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, no. January, pp. 230–235, 2015.

[14] R. Plösch, J. Bräuer, M. Saft, and C. Körner, "Design debt prioritization," in *Proceedings of the 2018 International Conference on Technical Debt - TechDebt '18*, vol. 10, no. 18. ACM, 2018, pp. 95–104.

[15] B. Ojameruaye and R. Bahsoon, "Systematic Elaboration of Compliance Requirements Using Compliance Debt and Portfolio Theory," in *20th International Working Conference, REFSQ, Essen, Germany*, 2014, pp. 152–167.

[16] J.-L. Letouzey, "The SQALE Method for Managing Technical Debt," in *Proceedings of the 3rd International Workshop on Managing Technical Debt (MTD12), IEEE, Zurich, Switzerland, 2012*, 2012, pp. 31–36.

[17] H. Markowitz, "Portfolio Selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[18] M. Engels, "Portfolio Optimization: Beyond Markowitz," Ph.D. dissertation, Universiteit Leiden, 2004.

[19] I. Omisore, M. Yusuf, and N. I. Christopher, "The modern portfolio theory as an investment decision tool," *Journal of Accounting and Taxation*, vol. 4, no. 2, pp. 19–28, 2012.

[20] J. Norstad, "An Introduction to Utility Theory," 2011. [Online]. Available: http://sriharisheshadri.com/docs/how-much-would-you-bet/norstad_utility.pdf

[21] C. Carlsson, R. Fullér, and P. Majlender, "A possibilistic approach to selecting portfolios with highest utility score," Turku Centre for Computer Science, Tech. Rep., 2002.

[22] S. Lim, T. J. V. Saldanha, S. Malladi, and N. P. Melville, "Theories Used in Information System Research : Insights from Complex Network Analysis," *Journal of Information Technology Theory and Application*, vol. 14, no. 2, pp. 5–46, 2013.

[23] I. Attarzadeh and S. H. Ow, "Project Management Practices: The Criteria for Success or Failure," *Communications of the IBIMA*, vol. 1, no. 28, pp. 234–241, 2008.

[24] The Open Group, "TOGAF." [Online]. Available: http://www.opengroup.org/togaf

[25] A. Ampatzoglou, A. Ampatzoglou, A. Chatzigeorgiou, and P. Avgeriou, "The financial aspect of managing technical debt: A systematic literature review," *Information and Software Technology*, vol. 64, pp. 52–73, 2015.

[26] M. G. Stochel, M. R. Wawrowski, and M. Rabiej, "Value-Based Technical Debt Model and Its Application," *ICSEA 2012, The Seventh International Conference on Software Engineering Advances*, no. c, pp. 205–212, 2012.