2014 IEEE 18th International Enterprise Distributed Object Computing Conference

Enterprise Architecture Analysis with Production Functions

Ulrik Franke
FOI – Swedish Defence Research Agency
SE-164 90 Stockholm, Sweden
e-mail: ulrik.franke@foi.se

***Abstract*—Enterprise Architecture (EA) is a discipline designed to cope with the complexity of modern enterprises at the intersection of information technology and business operations. This article demonstrates how EA models can be enriched with the production function concept from microeconomics, enabling new and business relevant kinds of analysis. The approach is demonstrated through examples regarding growth strategies, architectural efficiency with changing relative prices, and strategies for high availability IT services.**

Keywords*—*Enterprise Architecture, production functions, business analysis

I. INTRODUCTION

Modern businesses are increasingly dependent on information technology (IT). Information systems (IS) are being used for sales, customer relations management, human resources management, business intelligence, production, logistics and supply chain management, to name but a few. As a consequence, decision-making in the modern business environment increasingly needs to take address a nested complex of processes, people and technology. Enterprise Architecture (EA) has been established as a discipline to cope with this complexity and offer a bird's-eye view of the enterprise somewhere at the intersection of information technology and business operations. As such, it does not replace other ways to describe the enterprise and its operations, but rather complements them, just like a city plan does not replace technical drawings of individual buildings.

Furthermore, enterprise decision-makers are increasingly concerned with the IT/business interface. In a 2006 survey among chief information officers (CIOs), the most prioritized concerns were (i) to decrease the cost related to the business organization, (ii) to improve the quality of the interplay between the IT organization and the business organization and (iii) to provide new computer-aided support to the business organization [1]. The influential Gartner consultancy identifies a trend where business leaders are growing impatient with the ability of the IT organization to meet demand for IT-services and recommend that CIOs work with business teams to show the benefit of IT practices in operations [2]. At the same time, they point to the possibility of using enterprise architecture to drive business change by enabling CIOs to focus on the right business strategic priorities [3].

Another general trend is the increasing attention being given to quantitative analysis. Businesses that adopt 'data driven decision making' have been shown to enjoy output and productivity that is $5-6\%$ larger than their competitors [4]. The struggle to use big data in business decision-making [5] can also be seen through this lens.

This article aims to tie these two strands together by showing how EA models can be enriched with quantitative analysis at the intersection of IT and business. More specifically, we integrate the production function concept from microeconomics into EA analysis; showing how such analysis can be fueled by information from EA models, enabling better decision-making. The procedure is illustrated with a number of realistic CIO decision-problems.

A. Outline

The rest of this paper unfolds as follows: Section II introduces the basic production function concepts, followed by a brief note on modeling formalisms in Section III. The main contribution then ensues in Section IV, where a number of business cases are analyzed using the EA models enriched with production function concepts. Section V discusses the possibilities and limitations of the contribution, and Section VI puts it in context by discussing related work. Section VII concludes the paper.

II. PRODUCTION FUNCTIONS

There are many good introductions to production functions. In this article, we follow Varian when introducing the basic concepts [6].

Consider a firm that produces an output good y using factor inputs $x_1, x_2, \dots x_n$. The output might be cars, say, and the factor inputs might be labor, capital and raw materials. A production function is $f(x_1, x_2, \dots x_n) = y \in \mathbb{R}$ such that y is the maximum output associated with spending $x_1, x_2, \dots x_n$. (It is clear that we need to use the maximum output – it is always trivially possible to produce less than the maximum, simply by producing a bit of the output good and then wasting the rest of the factor inputs.) It is usually a good idea to think of inputs and outputs in terms of flows per time unit (whether day, month or year). The production function is used to summarize the production possibilities of a firm; succinctly expressing which combinations of inputs and outputs that are technologically feasible in the short run. (In the long run, new technologies offer new possibilities.)

A cost-minimizing firm producing y has to solve the

1541-7719/14 $31.00 © 2014 IEEE
DOI 10.1109/EDOC.2014.17

following constrained optimization problem:

$$\begin{aligned} &\min_{\mathbf{x}} \sum_{i=1}^{n} w_i x_i \\ &\text{such that } f(x_1, x_2, \ldots x_n) = y \end{aligned} \tag{1}$$

where $w_1, w_2, \ldots w_n$ are the prices of factor inputs $x_1, x_2, \ldots x_n$, respectively.

The n first order conditions for an optimal solution to this problem can be found using Lagrange multipliers. Dividing the i:th and j:th conditions yields the following condition, where we denote an interior solution point $(x_1^*, x_2^*, \ldots x_n^*) \in \mathbb{R}^n$ by the vector $\mathbf{x}^*$:

$$\frac{w_i}{w_j} = \frac{\frac{\partial f(\mathbf{x}^*)}{\partial x_i}}{\frac{\partial f(\mathbf{x}^*)}{\partial x_j}} \tag{2}$$

This condition is intuitively reasonable. The left hand side is the *economic rate of substitution* (ERS) – the rate at which factor j can be substituted for i maintaining a constant cost. The right hand side is the technical rate of substitution (TRS) – the rate at which factor j can be substituted for i maintaining constant level of output. Thus, the condition is that the ERS and the TRS are the same. If this were not the case, e.g. if the ERS were greater than the TRS then factor i could be substituted for factor j entailing smaller costs for the same level of output, meaning that $\mathbf{x}^*$ was in fact not optimal.

Many functional forms for production functions have been proposed, e.g. the Cobb-Douglas production function, the Leontief production function, the constant elasticity of substitution (CES) production function, and others. In this paper, we focus on the Cobb-Douglas function, since it has been widely used in studies of IT productivity. In general, the Cobb-Douglas function is written as follows:

$$f(\mathbf{x}) = k x_1^{\beta_1} x_2^{\beta_2} \ldots x_n^{\beta_n} \tag{3}$$

If $\sum_{i=1}^{n} \beta_i = 1$, the production function has constant returns to scale, i.e. doubling the factor inputs $x_1, x_2, \ldots x_n$ will also double output. If $\sum_{i=1}^{n} \beta_i < 1$, returns to scale are decreasing, and if $\sum_{i=1}^{n} \beta_i > 1$ returns to scale are increasing. The β parameters are known as the *output elasticities* of the factor inputs.

In a widely cited paper, Hitt and Brynjolfsson estimate output elasticities in a Cobb-Douglas model for the productivity of 370 firms [7]. Three factor inputs are used: total IT stock C, non-computer capital K and labor L, yielding the following Cobb-Douglas model for value added V (econometric dummy variables removed):

$$V = C^{\beta_1} K^{\beta_2} L^{\beta_3} \tag{4}$$

To create a single mesaure of IT – the IT stock – Hitt and Brynjolfsson aggregate two components: Computer capital, i.e. the total value of central processors and PCs owned by the firm, and IS labor, i.e. the labor portion of the IS budget. Based on the assumption that IS labor produces assets that last, on average, three years, the two components are combined thus:

$$\text{IT stock} = \text{IS Capital} + 3 \cdot \text{IS Labor} \tag{5}$$

Econometric analysis of the empirical data yields estimates of the output elasticities. Several estimates are discussed in the article, but we use the ordinary least squares (OLS) estimates reprinted in Table I throughout the rest of this article. (A recent meta-study finds a slightly increasing trend in IT productivity over time, but also a large variability over different studies [8]: numbers from other studies could easily be used.)

TABLE I. OUTPUT ELASTICITIES FOR TOTAL IT STOCK, NON-COMPUTER CAPITAL AND LABOR, FROM [7].

β_1	β_2	β_3
0.0883	0.212	0.663

One consequence of the Cobb-Douglas model is that the *budget shares* of the production factors are constant, regardless of relative prices (though of course the amount of factors that these budget shares will buy varies). This is precisely the fixed ERS in the left hand side of Eq. (2). Gurbaxani et al. test this consequence empirically on a six year data set of 138 Fortune 500 companies, and find that hardware and personnel factor shares remain constant over time, implying that information systems productivity can be characterized by Cobb-Douglas production [9]. Importantly, this analysis also shows that the Cobb-Douglas model is applicable not only to the aggregate economy level, but also to individual firms. This is confirmed by Cardona et al., listing a large number of firm level studies [8]. We now proceed to illustrate how the theory of production functions can be applied to firm level decision-making using enterprise architecture analysis methods.

III. MODELING FORMALISMS

This article uses the Predictive, Probabilistic Architecture Modeling Framework (P^2AMF) to perform calculations on the architecture examples in next section [10]. P^2AMF is designed to enable advanced and probabilistically sound reasoning about business and IT architecture models, given in the form of Unified Modeling Language (UML) class and object diagrams. P^2AMF is based on the Object Constraint Language (OCL), but also adds probabilistic inference. Furthermore, in this article the class and object diagrams are based on the ArchiMate language [11]. ArchiMate is a light-weight framework good for expositional purposes, since it can convey substantial IT and business information in a relatively small number of entities. The examples have been implemented in the Enterprise Architecture Analysis Tool (EAAT) [12], an academic tool for enterprise architecture modeling and analysis and the native environment of P^2AMF. While full descriptions of P^2AMF, ArchiMate and EAAT are beyond the scope of this paper, more details can be found in the work cited above. The contents of this paper, however, should be reasonably accessible without detailed knowledge of these modeling formalisms.

IV. BUSINESS CASE ANALYSIS

In this section, we use the proposed method to analyze a number of thought experiment business cases. Though the cases are fictitious, they give an idea of how the use of production functions can make a useful contribution to decision-making based on enterprise architecture analysis.

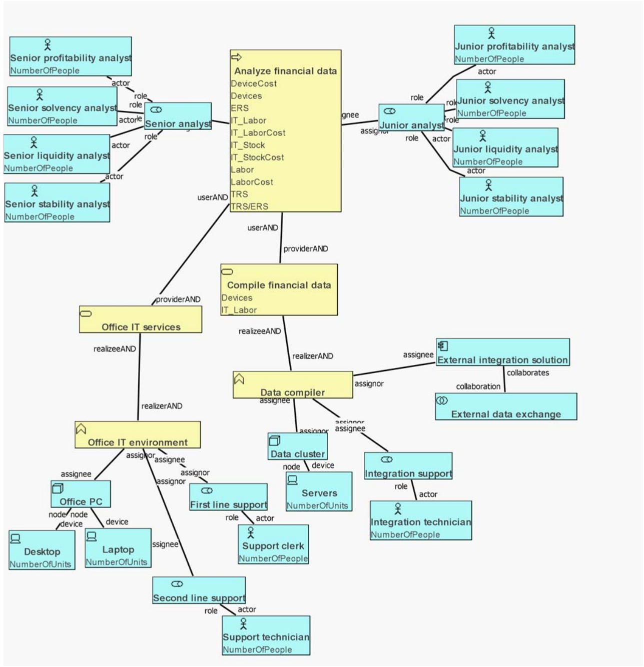

Fig. 1. The as-is enterprise architecture of the financial firm.

A. Extensive or intensive growth strategies

Consider the case of a financial analysis firm. As opposed to traditional factories, it does not rely very heavily on machinery, i.e. the traditional role of capital K. Rather, its human capital L and its information technology assets C constitute the bulk of its production. In the as-is case, it employs 300 financial analysts served by an IT department of 13, as illustrated in Fig. 1. Thinking about growth strategies, the board has prepared two alternatives; one for extensive and one for intensive growth. The extensive alternative involves expanding operations to 400 analysts, and correspondingly growing the IT department to 15 people in order to cope with increased need for support. As the new recruits to the IT department are mainly first-line support, the average IT labor cost will go down. The intensive alternative involves getting rid of 50 junior analysts, who mainly do data shuffling and simple analyses, and replace them with IT resources in the form of a Business Intelligence cluster. As the number of people supported is scaled down, so is the IT department, but new highly qualified IT labor is needed to maintain the new cluster. On a balance, the IT department will shrink to 10 people, but the average IT labor cost will go up. Now, how should the chief information officer (CIO) evaluate these perspectives from horizon of the IT department?

The two growth strategies are basically about re-adjusting the production factor of IT-stock C and labor L. Applying the conditions expressed in Eq. (2) to the model for V specified by Hitt and Brynjolfsson in Eq. (4), we have:

$$\frac{w_C}{w_L} = \frac{\frac{\partial V}{\partial C}}{\frac{\partial V}{\partial L}} = \frac{\beta_1 C^{\beta_1 - 1} K^{\beta_2} L^{\beta_3}}{\beta_3 C^{\beta_1} K^{\beta_2} L^{\beta_3 - 1}} = \frac{\beta_1}{\beta_3} \frac{L}{C} \tag{6}$$

Now, using the β-estimates from Table I, it is possible to evaluate any proposed architecture to see whether it satisfies the first order conditions. Indeed, given that the labor, IT labor and IT hardware is modeled in the enterprise architecture – as is the case Fig. 1 – Eq. (6) can be automatically evaluated. The CIO does not have to master microeconomics to use it for decision-support.

For the sake of clarity, let us walk through the attributes and relations of Fig. 1. The core business is the process `Analyze financial data`. This is carried out by people employed in different roles; `Junior analysts` and `Senior analysts`. Concretely, these are modeled as actors; `Junior profitability analyst`, `Senior liquidity analyst`, etc. To avoid having to explicitly model all individual actors, the attribute `NumberOfPeople` is set to reflect the number of people currently employed. The business process is supported by two technology services: the infrastructure service `Office IT services` and the application service `Compile financial data`. Looking into how these are realized, we find that the functions `Office IT environment` and `Data compiler` each are assigned technology nodes – `Office PC` and `Data cluster` – and roles – `First line support`, `Second line support`, and `Integration support`, respectively. At the leaves of the graph are devices and actors that concretely fulfill these roles and nodes. Just like the business actors, IT actors such as `Support technician` and devices such as `Laptop` have attributes (`NumberOfPeople`, `NumberOfUnits`) allowing convenient modeling of their numbers.

Returning to the core business process `Analyze financial data`, it has been endowed with a number of attributes necessary for the production function analysis. Some need to be specified by the CIO: the cost attribute for labor is just w_L, whereas those for devices and IT labor are used to calculate w_C as an average weighted by the number of devices and IT labor. These attributes, along with that for labor, are automatically calculated by traversing the architecture model. Thus, to determine that the `IT_Labor` should have the value 13, the dependencies of `Analyze financial data` are tracked down, so that the 6 `Support clerks`, 4 `Support technicians` and 3 `Integration technicians` are summed. In the case of `IT_Stock`, it is calculated as per Eq. (5), by adding `Devices` to three times `IT_Labor`. The ERS and TRS attributes are now straightforwardly calculated as the left hand side and right hand side of Eq. (6), respectively. To give an idea of what the corresponding P^{2}AMF code looks like, the following snippet shows the calculation of the TRS:

```
let
beta1=0.0883 --IT_stock
in
let
beta3=0.663 --Labor
in
beta1/beta3*self.Labor/self.IT_Stock
```

Returning to the business case, we can now calculate the TRS/ERS ratio – which should be unity when the firm operates optimally – for the as-is case and the two to-be cases, respectively. All that is required is to make the requisite changes in the architecture model, and let the EAAT tool recalculate. The architecture model for the intensive to-be case – with the added Business Intelligence cluster – is shown in Fig. 2. (The architecture model for the extensive to-be case differs from the as-is case only in the numbers set on key attributes, and is not shown.) For brevity, the results of the

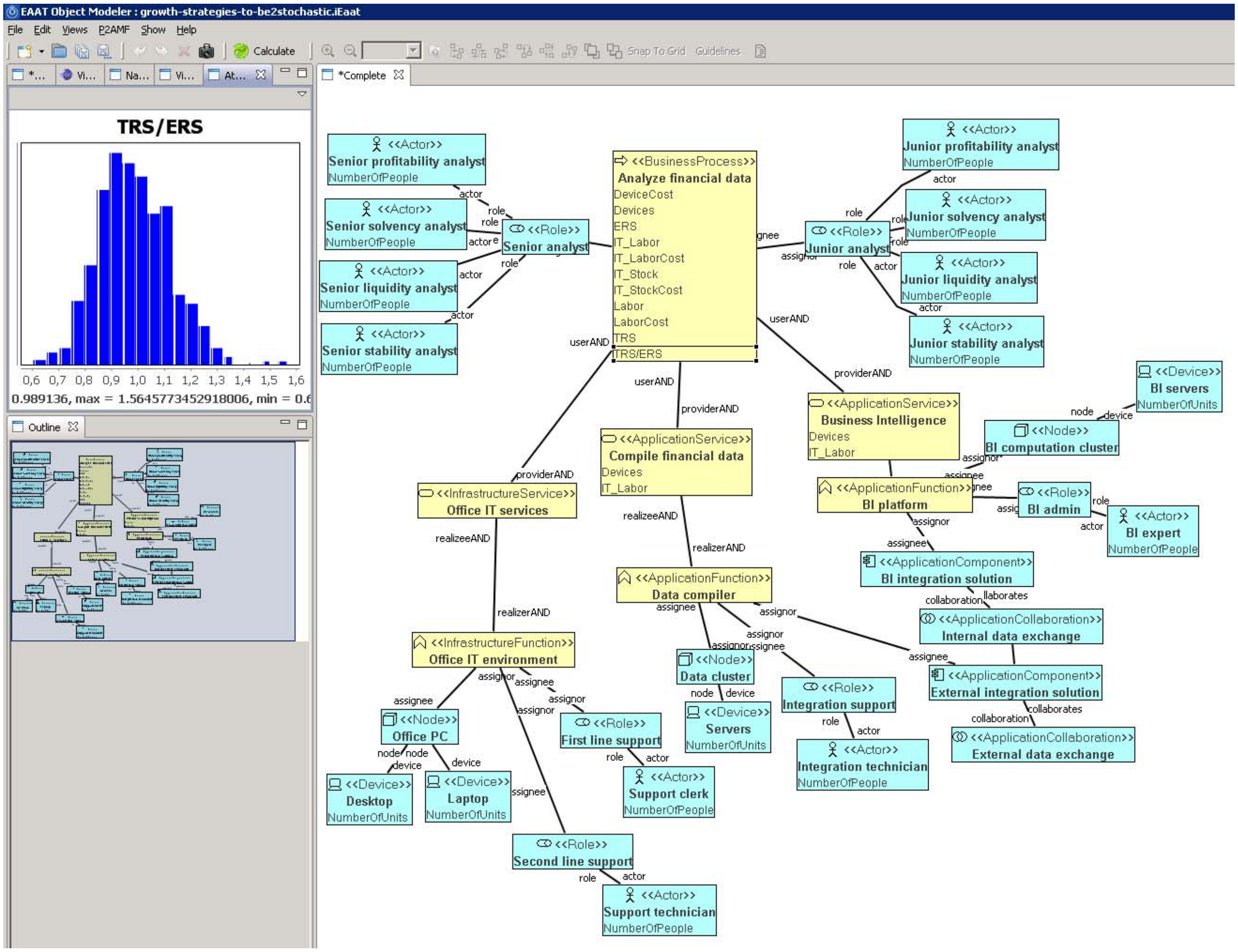

Fig. 2. The intensive to-be enterprise architecture of the financial analysis firm. Note the new Business Intelligence cluster, compared to Fig. 1. The histogram to the left shows the distribution of the TRS/ERS ratio for the stochastic case described in the text. As per the first order condition in Eq. (6), the firm should try to operate where this ratio is unity.

calculations are shown in Table II, with per annum costs.

TABLE II. A NUMERICAL COMPARISON OF THE THREE SCENARIOS FOR THE FINANCIAL ANALYSIS FIRM. ALL COSTS ARE PER ANNUM AND PER UNIT, AND INTENDED TO BE BALLPARK-REALISTIC IN \$ OR €. FOR THE IT DEVICES, THIS SHOULD BE A DEPRECIATED PRICE, E.G. THE LIST PRICE DIVIDED BY THE NUMBER OF YEARS IN USE.

	As-is	Extensive to-be	Intensive to-be
Labor	300	400	250
Devices	330	430	330
IT Labor	13	15	10
IT Stock	369	475	360
LaborCost	37000	37000	45000
DeviceCost	400	250	300
IT LaborCost	40000	38000	45000
IT StockCost	4585	3826	4025
TRS	0.108	0.112	0.092
ERS	0.124	0.103	0.089
TRS/ERS	0.87	1.08	1.03

Now, this enterprise architecture analysis helps solve the decision-problem of our CIO. First, by looking at the TRS/ERS quotient – unity when the first order conditions are met – it is clear that both to-be architectures (1.08 and 1.03) are more economically efficient than the as-is case (0.87). In this respect, both strategies make sense. Second, the intensive alternative is slightly better than the extensive, so all else being equal, this is the alternative that our CIO ought to recommend.

However, this result comes with some caveats. In particular, while the as-is labor costs are known, it would be appropriate to model the to-be salaries with stochastic variables for two reasons: First, it reflects uncertainty about future salary negotiations with the staff hired. Second – more importantly from the production economics perspective – it also reflects uncertainty about how much value will be delivered by each person. Even when someone has been hired and the salary settled, the value being delivered by that person to some extent remains an open question. A similar argument can be made for the cost of IT hardware. Fortunately, the probabilistic nature of the P^{2}AMF language allows us to model these uncertainties very easily.

The histogram to the left in Fig. 2 shows the results (for the intensive to-be case shown in the model) of letting

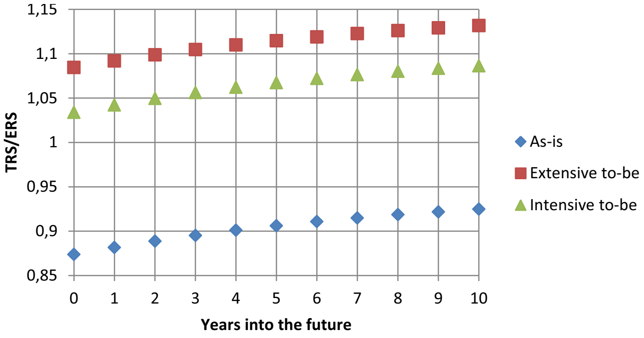

Fig. 3. The development of the TRS/ERS ratio over time, for the different architectures of the financial analysis firm.

the device cost be normally distributed with mean 300 and standard deviation 20, while both labor and IT labor costs are normally distributed with mean 45 000 and standard deviation 3 000. (The simulation was made with rejection sampling using 1 000 samples.) Perhaps, this is where our CIO gets the most powerful decision-support out of the model; allowing simple comparisons of the strategies under different cost assumptions.

B. Architectural efficiency with changing relative prices

The efficiency of a given architectural solution is not only affected by managerial decisions, as illustrated by the growth strategies in the previous example. It is also gradually affected by everyday phenomena such as changing relative prices. As pointed out by Gurbaxani et al., in a Cobb-Douglas model the optimal budget shares of input factors such as personnel and hardware should remain the same over time [9]. But keeping this ratio constant will require conscious management decisions in the realistic case where personnel gets more expensive and hardware gets less expensive over time. Indeed, enterprise IT systems are relatively long-lived – very roughly, their half-lives might be in the decade range [13]. Therefore, the issue of efficiency in the sense of Eq. (2) should not only be assessed in a single year, but over time.

Fortunately, the enterprise architecture analysis approach illustrated in the previous section extends in a straightforward manner to the case of changing relative prices. Looking again at the case of the financial analysis firm, the CIO can easily change the cost attributes of the business process (`DeviceCost`, `LaborCost`, `IT_LaborCost`) to reflect beliefs in the future development of prices. The EAAT tool automatically calculates the updated TRS/ERS ratio.

If desired, the problem can be assessed systematically under different assumptions. For example, if labor (financial analysts and IT staff alike) cost is expected to grow by 4% per annum, and device costs are expected to shrink by 8% per annum, the as-is, extensive to-be and intensive to-be cases will develop as illustrated in Fig. 3 over the next decade. From an economic efficiency perspective, such as development would make the as-is case slightly better, and both to-be cases slightly worse, as they move closer and further from the unity TRS/ERS ratio, respectively. It is worth noting that the reason why the TRS/ERS ration for IT Stock and Labor does not change more is the large component of IT Labor – following the same cost trajectory as Labor – in the IT Stock concept, as expressed in Eq. (5).

C. Strategies for high availability IT services

Consider the case of a business critical IT service with high requirements on availability. The service might be part of an industrial process (e.g. paper manufacturing) or a business process (e.g. trading on the stock market) where downtime entails large costs, or it might be part of some critical infrastructure (e.g. air traffic control) where downtime poses significant risks to human life.

To make matters more concrete, consider an e-commerce firm, where the availability of the electronic sales application service is essential (this is where the revenue is made, and this is where customers are lost if service is unavailable). Business operations impose a demand for an availability level of 99.81% – the level corresponding to ”continuous availability” according to renowned analysis firm Gartner [14]. How should the CIO meet this requirement?

Steady state availability (roughly mean uptime over a time interval) is defined mathematically as follows:

$$A = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}} \tag{7}$$

MTTF denotes ”Mean Time To Failure” and MTTR ”Mean Time To Repair” or ”Mean Time To Restore”. As *mean* times are used, Eq. (7) measures the long-term performance of a system. Now, if we consider availability the output good, the definition suggests a model with two factor inputs [15]: Capital K can buy better hardware (or more of the same, to build redundancy), increasing the MTTF; Labor L can be used to monitor the system and take swift action if it fails, decreasing the MTTR. While, to my knowledge, production functions have not previously been applied to IT service availability, the Cobb-Douglas model has been used in studies of software reliability [16]. Inspired by this, a simple but not unreasonable model posits the following Cobb-Douglas-like functional forms for MTTF and MTTR, respectively:

$$\text{MTTF} = k_2 K^{\beta_2} \tag{8}$$

$$\text{MTTR} = k_3 L^{-\beta_3} \tag{9}$$

(The parameter indices 2 and 3 are chosen to be aligned with Eq. (3).) Note the minus sign in Eq. (9) – MTTR decreases as more labor is added.

Applying the conditions expressed in Eq. (2) to the availability model defined by Eqs. (7), (8) and (9), we obtain, after some algebra:

$$\frac{w_K}{w_L} = \frac{\frac{\partial A}{\partial K}}{\frac{\partial A}{\partial L}} = \frac{\beta_2}{\beta_3}\frac{L}{K} \tag{10}$$

With Eq. (10) it is possible to evaluate whether proposed IT service architectures are efficient with respect to availability. Thus, if the CIO obtains architecture proposals from staff or contractors, they can in principle be evaluated with Eq. (10), much like the previous example with the financial analysis firm. However, the model proposed in Eqs. (8) and (9) also makes it possible to generate proposals. To find the cheapest

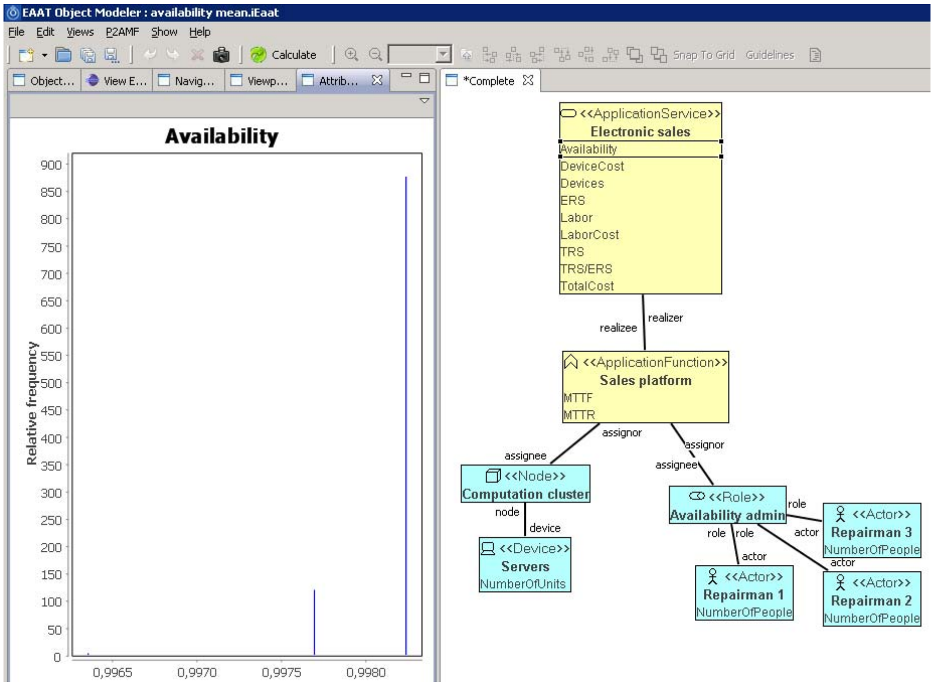

Fig. 4. Availability analysis for the e-commerce case. The x-axis shows different availability levels.

way of achieving a desired availability level A^*, the following cost minimization problem has to be solved:

$$\begin{aligned} &\min_{L,K}\ w_L L + w_K K \\ &\text{such that } \frac{k_2 K^{\beta_2}}{k_2 K^{\beta_2} + k_3 L^{-\beta_3}} = A^* \end{aligned} \tag{11}$$

Solving the constraint for each factor allows us to write the expression to be minimized in terms of only one factor and A^*. The first-order conditions for the minimization problem are then found by finding where the derivatives with respect to each of the factors are zero. This leads to the following conditional demand functions for each factor:

$$L^*(w_L, w_K, A^*) = \left(\frac{w_L}{w_K}\frac{\beta_2}{\beta_3}\right)^{-\frac{\beta_2}{\beta_2+\beta_3}} \left(\frac{A^*}{1-A^*}\frac{k_3}{k_2}\right)^{\frac{1}{\beta_2+\beta_3}} \tag{12}$$

$$K^*(w_L, w_K, A^*) = \left(\frac{w_L}{w_K}\frac{\beta_2}{\beta_3}\right)^{\frac{\beta_3}{\beta_2+\beta_3}} \left(\frac{A^*}{1-A^*}\frac{k_3}{k_2}\right)^{\frac{1}{\beta_2+\beta_3}} \tag{13}$$

(Of course, we see that dividing L^* with K^* meets the optimality condition expressed in Eq. (10).)

The cheapest way of achieving 99.81% availability is given by Eqs. (12) and (12). For the sake of the example, we use β_2 and β_3 from Table I, set $k_2 = 20\ 000$, $k_3 = 200$, $w_L = 40\ 000$ and $w_K = 300$. Rounded to whole numbers, we find $K^* = 115$ and $L^* = 3$, i.e. a server cluster corresponding to 115 'normal' devices should be maintained by a service crew of 3. This solution architecture is depicted to the right in Fig. 4.

However, this is not the whole truth. Again, there is uncertainty to be managed. In particular, while the cluster remains constant following the initial investment, the number of people in the repair crew might vary due to vacancies; e.g. initial recruitment problems or people resigning. If our CIO is prudent, she might consider the consequences of such manpower shortage. Fig. 4 shows the results of assuming that each of the 3 persons in the repair crew has a 95% chance of actually being available (such a figure could be set based on experience of employee turnover rates). In P^{2}AMF terms, their `Role-Actor` relations are assigned 95% probabilities of existence. The availability diagram to the left shows the relative frequencies for different outcomes – most often all 3 repairmen are there, yielding the desired 99.81% availability, but sometimes one or even two are absent, resulting in lower numbers. (Luckily, the case of three absentees is so improbable – $0.05^3 < 0.001$ – that it does not show up among our $1\ 000$ samples.)

Another important limitation is hidden in the use of *mean* times to failure and repair. Such mean values can be misleading [17]. A more realistic model would replace the bare means with full distributions – such as the Weibull distribution for time to failure (TTF) [18] and the log-normal distribution for time to repair (TTR) [19]. Though single mean values cannot uniquely determine such two parameter distributions, it is still possible to pick illustrative Weibull TTF and log-normal TTR distributions, consistent with the mean values determined through the production function calculations above. Such an analysis can bring additional insight to our CIO. Fig. 5 illustrates the result of modeling full distributions in the EAAT tool, clearly illustrating that meeting an availability target on average does not necessarily amount to meeting it every time. We note that though most outcomes are close to the availability target set, some outcomes show up at lower levels (even outside the diagram; the axis has been truncated). While this gives realistic feeling for the probabilistic nature of availability outcomes, it should also be noted that the analysis

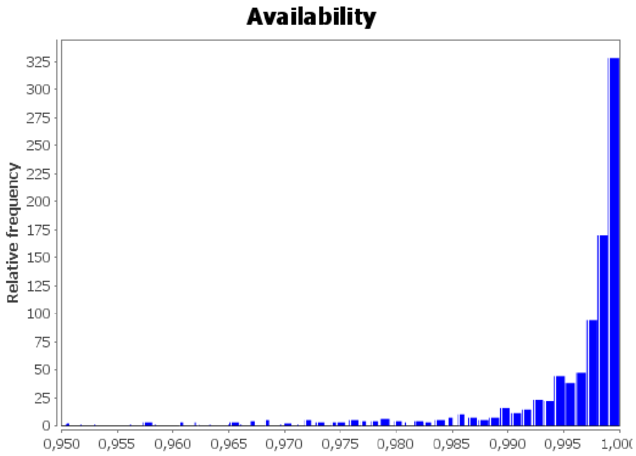

Fig. 5. Availability analysis for the e-commerce case, with Weibull TTF and log-normal TTR. The x-axis shows different availability levels.

depicted in Fig. 5 is a bit pessimistic. This is because each availability sample corresponds to picking a *single* outcome from each of the TTF and TTR distributions. Typically, steady state availability should be measured and evaluated over a period of time with *several* outages, each of which should involve renewed TTF and TTR samples, smoothing away some extreme outcomes.

V. DISCUSSION

The examples in the preceding section are constructed to show different aspects of how production functions can be used in EA analysis. The case of the financial firm growth strategies has two main points. First, it illustrates how existing research can be put to use. Here, the model used builds on the 370 firms econometrically analyzed by Hitt and Brynjolfsson [7]. Surely, such data could be put to good use in enterprise decision-making – at least when aided by the proper tools. Second, it illustrates the use-case where a proposed technical solution already on the table is evaluated from the point of view of economic efficiency. It is reasonable so suppose that this is the most common case when it comes to overall business strategy – mergers and acquisitions, new product launches, or large organizational changes. Here, the IT/business interface does not determine the options, but can still have a well-founded say about the choices.

The case of the high availability strategies is deliberately chosen to illustrate two other points. First, it illustrates the case when there is no single strand of research that can solve the entire business case, but rather assumptions have to be made (about the Cobb-Douglas-like production function for availability, about manpower shortage, about statistical distributions). Of course, such assumptions sometimes have to be made, regardless of analysis model. This is not unique for this method. However, expressing all of these concerns in a single EA model can be beneficial for the holistic view sought by the decision-maker. Second, it illustrates the use-case where an optimal technical solution is generated from the microeconomics. While this is an unlikely use-case for overall business strategy, it is a more likely use-case when the scope is entirely within the domains of the IT department. Still, it is important to remember that production functions are black-box abstract models of technology – not blueprints for particular solutions.

The issue about blue-prints vs. black boxes and the need for making assumptions in early stages relates to the life cycle perspective. EA analysis about to-be scenarios is largely about managing the uncertainty of future states, even with limited information. The INCOSE Systems Engineering Handbook illustrates how the cost to extract defects grows as a project progresses – correcting errors in the design phase is 3-6 times more costly than in the concept phase, but that cost grows to 20-100 times in the development phase and 500-1 000 times in the production and testing phases [20]. In a sense, microeconomics adds another tool to the EA toolbox for finding defects as early as possible.

The examples in the previous section do not share a common metamodel as such. As indicated in Section III, we have loosely adhered to ArchiMate, but without loss of generality. However, a number of attributes are central to the production function approach illustrated: `ERS` and `TRS` for the production factors considered, their ratio `ERS/TRS`, the prices of the factors (e.g. `LaborCost` and `IT_StockCost`) and the quantities of the factors (e.g. whole numbers for `Devices` and `Labor`). These need to be included to enable the analysis. Furthermore, the factors should be calculable from the architecture model itself, e.g. traversing departments, processes, etc. to count the employees in various roles.

VI. RELATED WORK

EA models can be used for many purposes, e.g. (i) documentation and communication, (ii) analysis and explanation, and (iii) design [21]. In this article, we are primarily concerned with the second and third uses. There is a lot of previous literature in this field. For example, advanced EA analysis models have been proposed for modeling and assessing technical systems qualities such as availability [22], security [23], interoperability [24] and modifiability [25]. While very related in methodology, these works are all concerned with modeling the concerns of the IT department in a way making it business relevant. The use of microeconomics in this article, as a contrast, adopts a more holistic perspective, where IT is just one production factor among others, such as labor.

Other work within the EA for design strand addresses EA management [26] and EA patterns [27], business-IT alignment [28], essential modeling practices [29] and the state of the EA practice [30]. All of these works share the goal of this paper – using EA as a tool to improve business operations – but none adopts the microeconomics perspective. Indeed, it seems that this perspective has not previously been integrated into the EA analysis paradigm.

VII. CONCLUSION

This article has demonstrated how enterprise architecture models can be enriched with the production function concept from microeconomics. By extending models in this way, a number of business analysis concerns can be addressed, using information already in the EA models, along with a few additional assumptions. The approach has been demonstrated with regard to growth strategies, architectural efficiency with changing relative prices, and strategies for high availability IT services.

There are several candidates for interesting future work. First, from a theoretical point of view, it would be interesting to catalog the existing studies that – like [7] – could be useful in creating EA analysis models. Second, from a practical point of view, it would be interesting to try the methods outlined in the paper on real enterprise decision-making. Third, from an availability engineering point of view, it would be interesting to test the Cobb-Douglas-like availability production function expressed in Eqs. (8) and (9) econometrically with empirical data.

ACKNOWLEDGMENT

This work was supported by the R&D program of the Swedish Armed Forces. Markus Buschle, Mika Cohen, Peter Hammar, Vahid Mojtahed and Daniel Oskarsson gave valuable comments on an early draft of the article.

REFERENCES

[1] Å. Lindström, P. Johnson, E. Johansson, M. Ekstedt, and M. Simonsson, "A survey on CIO concerns – do enterprise architecture frameworks support them?" *Information Systems Frontiers*, vol. 8, no. 2, pp. 81–90, 2006.

[2] S. Mingay, P. Iyengar, and K. Potter, "How CIOs Need to Prepare for Industrializing and Mutualizing IT in the Digital Economy," Gartner, Inc., Tech. Rep., Feb. 2014.

[3] M. Blosch and B. Burton, "Enterprise Architects Empower CIOs to Drive Business Collaboration With Business Capabilities," Gartner, Inc., Tech. Rep., Dec. 2013.

[4] E. Brynjolfsson, L. Hitt, and H. Kim, "Strength in numbers: How does data-driven decisionmaking affect firm performance?" *Available at SSRN 1819486*, 2011.

[5] H. Chen, R. H. Chiang, and V. C. Storey, "Business intelligence and analytics: From big data to big impact," *MIS Quarterly*, vol. 36, no. 4, pp. 1165–1188, 2012.

[6] H. R. Varian, *Microeconomic analysis, third edition*. W. W. Norton & Company, Inc. New York, 1992.

[7] L. M. Hitt and E. Brynjolfsson, "Productivity, business profitability, and consumer surplus: three different measures of information technology value," *MIS Quarterly*, pp. 121–142, 1996.

[8] M. Cardona, T. Kretschmer, and T. Strobel, "ICT and productivity: conclusions from the empirical literature," *Information Economics and Policy*, vol. 25, no. 3, pp. 109–125, 2013.

[9] V. Gurbaxani, N. Melville, and K. Kraemer, "The production of information services: a firm-level analysis of information systems budgets," *Information systems research*, vol. 11, no. 2, pp. 159–176, 2000.

[10] P. Johnson, J. Ullberg, M. Buschle, U. Franke, and K. Shahzad, "An architecture modeling framework for probabilistic prediction," *Information Systems and e-Business Management*, pp. 1–28, 2014.

[11] The Open Group, "Archimate 2.1 specification," 2013, available from https://www2.opengroup.org/ogsys/catalog/C13L.

[12] M. Buschle, P. Johnson, and K. Shahzad, "The enterprise architecture analysis tool–support for the predictive, probabilistic architecture modeling framework," in *19th Americas Conference on Information Systems, AMCIS 2013*, 2013, pp. 3350–3364.

[13] S. Aier, S. Buckl, U. Franke, B. Gleichauf, P. Johnson, P. Närman, C. M. Schweda, and J. Ullberg, "A survival analysis of application life spans based on enterprise architecture models." in *EMISA*, 2009, pp. 141–154.

[14] B. Malik and D. Scott, "How to Calculate the Cost of Continuously Available IT Services," Gartner, Inc., Tech. Rep., Nov. 2010.

[15] U. Franke, "Optimal IT Service Availability: Shorter Outages, or Fewer?" *Network and Service Management, IEEE Transactions on*, vol. 9, no. 1, pp. 22–33, Mar. 2012, DOI: 10.1109/TNSM.2011.110811.110122.

[16] P. Kapur, H. Pham, A. G. Aggarwal, and G. Kaur, "Two dimensional multi-release software reliability modeling and optimal release planning," *Reliability, IEEE Transactions on*, vol. 61, no. 3, pp. 758–768, 2012.

[17] A. Snow and G. Weckman, "What are the chances an availability SLA will be violated?" in *Networking, 2007. ICN'07. Sixth International Conference on*. IEEE, 2007, pp. 35–35.

[18] B. Schroeder and G. Gibson, "A large-scale study of failures in high-performance computing systems," *Dependable and Secure Computing, IEEE Transactions on*, vol. 7, no. 4, pp. 337–350, 2010.

[19] U. Franke, H. Holm, , and J. König, "The distribution of time to recovery of enterprise IT services," *Reliability, IEEE Transactions on*, 2014, in press.

[20] C. Haskins, K. Forsberg, M. Krueger, D. Walden, and D. Hamelin, *Systems engineering handbook*, 2006.

[21] S. Kurpjuweit and R. Winter, "Viewpoint-based meta model engineering," in *Enterprise Modelling and Information Systems Architectures – Concepts and Applications, Proceedings of the 2nd Int'l Workshop EMISA*, vol. 119, 2007, pp. 143–161.

[22] U. Franke, P. Johnson, and J. König, "An architecture framework for enterprise IT service availability analysis," *Software & Systems Modeling*, pp. 1–29, 2013.

[23] T. Sommestad, M. Ekstedt, and H. Holm, "The cyber security modeling language: A tool for assessing the vulnerability of enterprise system architectures," *Systems Journal, IEEE*, vol. 7, no. 3, pp. 363–373, 2013.

[24] J. Ullberg, P. Johnson, and M. Buschle, "A language for interoperability modeling and prediction," *Computers in Industry*, vol. 63, no. 8, pp. 766–774, 2012.

[25] R. Lagerström, P. Johnson, and D. Höök, "Architecture analysis of enterprise systems modifiability–models, analysis, and validation," *Journal of Systems and Software*, vol. 83, no. 8, pp. 1387–1403, 2010.

[26] S. Buckl, F. Matthes, and C. M. Schweda, "A viable system perspective on enterprise architecture management," in *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*. IEEE, 2009, pp. 1483–1488.

[27] S. Buckl, A. M. Ernst, F. Matthes, R. Ramacher, and C. M. Schweda, "Using enterprise architecture management patterns to complement togaf," in *Enterprise Distributed Object Computing Conference, 2009. EDOC'09. IEEE International*. IEEE, 2009, pp. 34–41.

[28] J. Luftman, "Assessing business-it alignment maturity," *Strategies for information technology governance*, vol. 4,

p. 99, 2004.

[29] R. Winter and R. Fischer, “Essential layers, artifacts, and dependencies of enterprise architecture,” in *Enterprise Distributed Object Computing Conference Workshops, 2006. EDOCW’06. 10th IEEE International*. IEEE, 2006, pp. 30–30.

[30] S. Aier, B. Gleichauf, and R. Winter, “Understanding enterprise architecture management design-an empirical analysis.” in *Wirtschaftsinformatik*, 2011, p. 50.