Business Architecture Quantified: How to Measure Business Complexity

13

Christian Schmidt

Abstract

Complexity of both business and IT is one of the most frequently discussed topics in strategic management and enterprise architecture today. For many business leaders, complexity is of central concern due to its assumed impacts on operating costs, organizational agility, and operational risks. In fact, complexity growth may be considered one of the major drivers for misalignment. As a consequence, organizations are increasingly forced to manage the complexity of their business and IT actively. However, existing qualitative methods fall short of supporting this on a larger scale. Quantitative measures may be considered a promising means to assess and manage the complexity of business and IT architectures in a systematic and universal way. This chapter presents a generic framework for conceptualizing and measuring enterprise architecture complexity and applies it to the domain of business architecture. Using this book's business architecture framework as a reference, it is shown how business complexity can be operationalized and quantified using well-defined and practice-proven measures.

13.1 Why Complexity Matters

Complexity is blamed for many things. Many business leaders seem to think of it as a general source of evil. Complexity is held responsible for rising coordination efforts and operating costs. Complexity is said to drive up change efforts, thereby constraining agility and swelling the time-to-market. And complexity is perceived a major source of failures, poor quality, and increasing operational risks. But there is also another side to the story. Against the backdrop of growing market dynamics,

C. Schmidt (✉)
Scape Consulting GmbH, Frankfurt, Germany
e-mail: christian.schmidt@scape-consulting.de

D. Simon, C. Schmidt (eds.), *Business Architecture Management*, Management for Professionals, DOI 10.1007/978-3-319-14571-6_13

competition, and legal regulation, organizations are facing a permanent need to develop new and innovative solutions. Often, this comes at the price of expanding business and information technology (IT) complexity only. US mutual insurer USAA, for example, has been reported to deliberately take up higher levels of complexity in order to create a high-quality customer experience (Mocker and Ross 2012). As it stands, complexity is a burden, but it may also be a necessity. This *Janus face* of complexity together with the lack of a commonly agreed definition is a major source of confusion, making complexity management a rather controversial and challenging subject. It is the purpose of this chapter to add some more clarity to the discussion and show how the concept can be applied to the domain of business architecture.

Generally speaking, complexity may be considered a quality of a system (or architecture) referring to the *quantity* and *variety* of system *elements* and the *relationships* between these (cf. Schütz et al. 2013; Schneberger and McLean 2003). Per se, complexity is neither a good nor a bad thing. But as a matter of fact, it has various implications for the development, change, and operation of the system. Therefore, complexity should be regulated to an appropriate level. But what exactly does that mean?

Fundamentally, each system/architecture needs to fulfill the requirements imposed to it by the environment.[1] Fulfilling these requirements will call for a certain *minimum level of complexity* that cannot be reduced without causing dysfunctional behaviour.[2] Any complexity exceeding this minimum level may in turn be considered *architectural waste*. Waste elements and relationships are problematic in the sense that they will increase operations, change, and maintenance efforts. In the following, architectural waste is also referred to as the *complexity surplus* of an architecture or parts thereof. According to this terminology, approaching the optimal level of complexity equates to minimizing the complexity surplus.[3] However, in order to identify the minimum complexity, the requirements must be known. This may be straightforward for a certain software application. But how does this concept relate to enterprise and business architecture?

In enterprise architecture management (EAM), architecture layers have emerged as a good practice to structure and decouple the main parts of the overall architecture. The ArchiMate standard, for instance, distinguishes between a business, application, and technology layer and uses the concepts of business, application, and technology services to decouple these (The Open Group 2013).

[1] In line with the classic dichotomy coined by Drucker (1974), an architecture that fulfills all environmental requirements may be called effective (it “does the right thing”).

[2] Referring to the classic dichotomy again, an architecture with minimal complexity may be called efficient (it “does the things right”, i.e., with minimal effort) (Drucker 1974).

[3] It should be noted though that the strategic impact of the complexity surplus will be contingent on the role of depending variables like agility and efficiency within the organization. For example, a quality leader operating in a stable market environment may have less incentives to control the complexity surplus than a cost leader in a rapidly developing marketplace.

Following this conception, the requirements of a given layer are defined by the layer above. The application architecture, for example, needs to fulfill the requirements of the business architecture by providing appropriate application services. Taking a closer look into the business architecture as conceptualized in Chap. 1, the business execution layer needs to satisfy the requirements imposed by the business model, and in turn, the business motivation. Therefore, to determine the complexity surplus of a given architecture layer or domain, the complexity requirements of the overarching layer need to be evaluated. Successful complexity management will hence be characterized by minimizing the complexity of each architecture layer taking into account the layers above. This may be considered a major strategy to achieve architectural consistency and alignment.[4] As each architecture layer inherits complexity requirements from the layer above, the management of complexity will be most effective if layers are addressed in a top-down order.

However, it should be noted that requirements (and thus the optimal level of complexity) may vary across vertical domains or capabilities (Schmidt 2013). For differentiating front-end domains, for instance, a higher level of complexity may be appropriate than for non-differentiating back-end domains. Therefore, effective complexity management is not simply about reducing complexity throughout the landscape but rather about creating the right level of complexity in the right place, that is, finding the right positioning in the *complexity continuum* as symbolized in Fig. 13.1.

What makes an active management of business and IT complexity even more important is the underlying dynamics known as the *law of rising complexity* (Schmidt and Buxmann 2011; Lehmann 1997; see Fig. 13.2). In order to survive, organizations constantly need to adapt to changing environmental conditions.

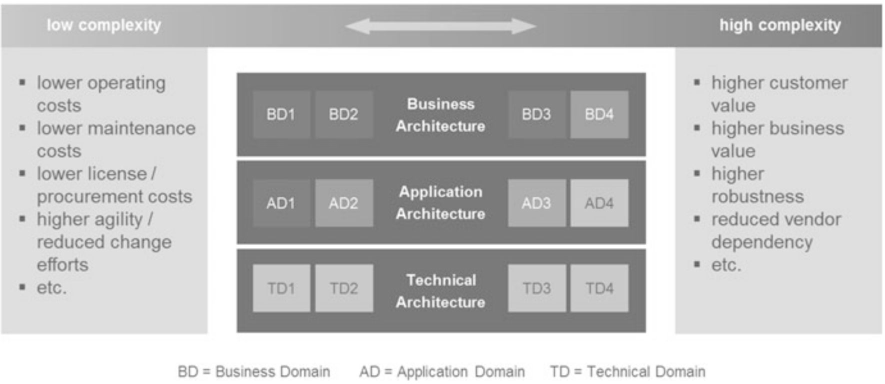

Fig. 13.1 The complexity continuum

[4] From an architectural perspective, alignment may be defined as the degree of consistency between the components of an architecture given by their properties and collocation. An architecture is well aligned if it is both effective (fulfills all requirements imposed by the environment) and efficient (does not contain any waste components or relationships).

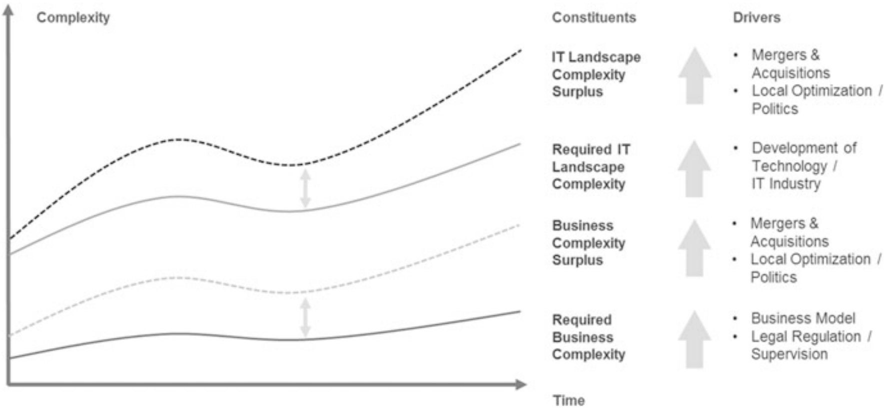

Fig. 13.2 The law of rising complexity (see also Rutz 2012)

In practice, this generally follows an evolutionary process mediated by internal and external stakeholders. This evolutionary process tends to favor local and short-term solutions, mirroring the need for swift implementation and the balancing of stakeholder power (Schmidt and Buxmann 2011). The managers of certain business lines, for example, often strive to create local solutions that they can control and shield from the rest of the organization. Also, mergers and acquisitions usually add to business and IT complexity. If not managed actively, complexity will hence rise continuously. Given the negative implications outlined above, an effective management of business and IT complexity may be considered a strategic capability that may even be turned into a source of sustained competitive advantage.

13.2 The Need for Quantitative Models

Given the importance of complexity in both business and IT architecture, methods are needed to actively manage and control the complexity within the various architectural domains. Until recently, complexity decisions have been mostly based on *qualitative reasoning*. Employing architectural repositories, enterprise architects usually engage in capturing and maintaining a structured model of the architecture (including components and their interrelationships). Traditionally, this data is primarily used for qualitative analyses. For example, graphical views and matrices may be created to demonstrate that certain capabilities have multiple (redundant) implementations or that key strategies are poorly supported by IT applications.

While this approach is working fine at the level of individual applications or even small landscapes, it has its limitations when it comes to very large business or IT landscapes as commonplace in today's multinational corporations. Practically, such architectures cannot be visualized graphically anymore. Also, the efforts required for a qualitative analysis may easily rise beyond the level feasible. Even more importantly, there are no proven mechanisms to aggregate the results of such qualitative analyses to a higher abstraction level (e.g., from domain to enterprise scale) and thus create a condensed high-level view.

To overcome these drawbacks, qualitative methods may be complemented by *quantitative methods* using dedicated *complexity measures*. Such measures could be calculated and aggregated across whole landscapes and integrated into a high-level reporting on the fundamental properties of an architecture.[5] The next section presents a generic framework that can be used to derive complexity measures in a systematic way. This is then applied to the domain of business architecture.

13.3 A Generic Framework for Measuring Complexity

Until today, no specific methods have been proposed to quantify the complexity of business architectures. However, measuring the complexity of enterprise architectures in general and IT architecture in particular has been approached by researchers more recently (see Mocker 2009; Widjaja et al. 2012; Schütz et al. 2013; Schmidt et al. 2013; Lagerström et al. 2013; Schneider et al. 2014a, b). In particular, a generic framework for conceptualizing and measuring enterprise architecture complexity has been proposed by Schütz et al. (2013) and further operationalized by Schmidt et al. (2013). In the following, the approach is presented and then extended to meet the requirements of a holistic complexity analysis.

13.3.1 The Heterogeneity-Based Complexity Model

According to the approach proposed by Schütz et al. (2013), the (structural) complexity of a system is defined along four dimensions: the *number* (or *quantity*) and *heterogeneity* (or *variety*) of system *elements* and *relations*. This approach is generic in the sense that it can be applied to any type of system and architecture including technical architecture, application architecture, and business architecture (cf. Fig. 13.3).

Following this approach, the problem of quantifying complexity is reduced to quantifying *heterogeneity*. In this context, heterogeneity (also referred to as *variety* or *concentration*) is defined as the diversity of elements or relationships of a system with respect to certain *characteristics* (attribute values). Heterogeneity can be captured as a statistical property and be described by means of empirical frequency distributions. For example, the distribution of database management systems within an IT landscape may be captured as shown in Fig. 13.4.

Based on such frequency distributions, statistical concentration measures may be applied (Widjaja et al. 2012). In particular, the *entropy measure* as introduced by

[5] In contrast to prevalent methods in the EAM field, this could be a key constituent of what may be called "*Quantitative EAM*."

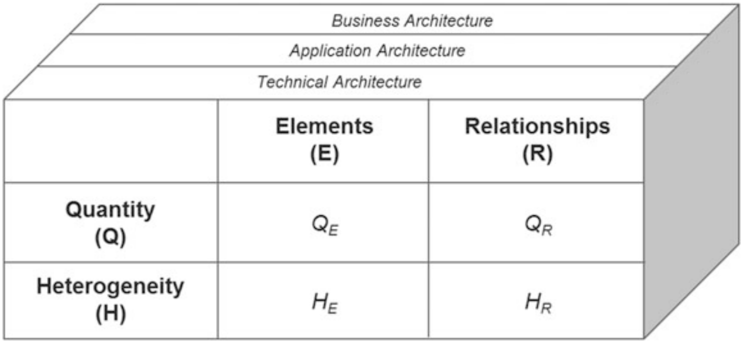

Fig. 13.3 Complexity dimensions (Schmidt 2013; see also Schütz et al. 2013)

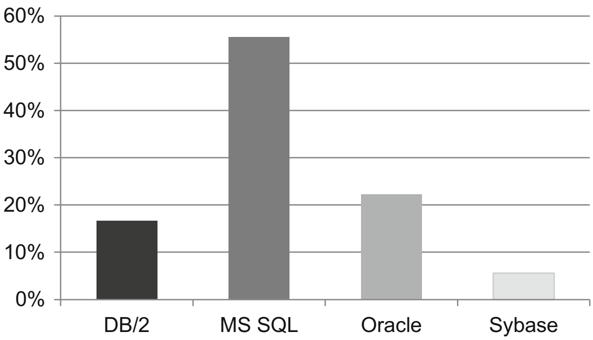

Fig. 13.4 Frequency distribution of database management systems by type

Shannon (1948) has been shown to be well suited to measure heterogeneity within enterprise architectures (Schütz et al. 2013). Formally, the entropy measure is defined as

$$EM = \sum_{i=1}^{n} f_i \ln\left(\frac{1}{f_i}\right)$$

with f_i denoting the relative frequency of the respective attribute values (characteristics). As shown in Fig. 13.5, the entropy measure increases with the number of different characteristics and with approaching an equal distribution. In contrast to similar measures, it is also sensitive with respect to characteristics with small shares. Yet, proportional changes of absolute frequencies have no impact on the measure.

The entropy measure takes its minimal value if all elements share a single characteristic. The maximum value is reached at equal distribution to different

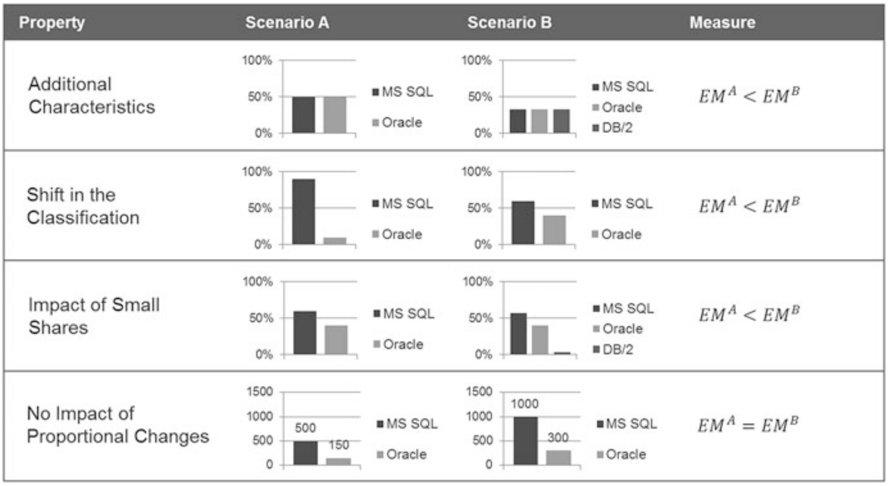

Fig. 13.5 Properties of the entropy measure (cf. Schütz et al. 2013)

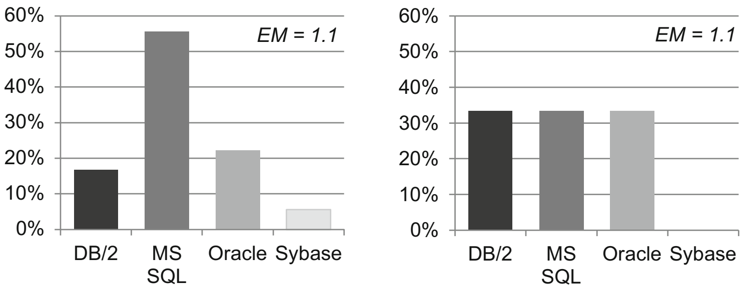

Fig. 13.6 Original distribution and equivalent equal distribution $\left(EM^* = exp(1.1)\ =\ 3\right)$

characteristics. Interpretation of the entropy measure is facilitated by the so-called *numbers equivalent entropy measure* $EM^* = exp(EM)$, which denotes the equivalent number of characteristics at equal distribution (see Fig. 13.6).

As shown in Schmidt et al. (2013), the generic heterogeneity-based approach may be used flexibly with any particular architecture framework and metamodel. In doing so, it may not only be applied to architecture elements but also to direct and indirect relationships between these (see Fig. 13.7). For example, the distribution of application systems along the underlying technology platforms or the concentration of business functions on applications may be analyzed (see Schmidt et al. 2013).

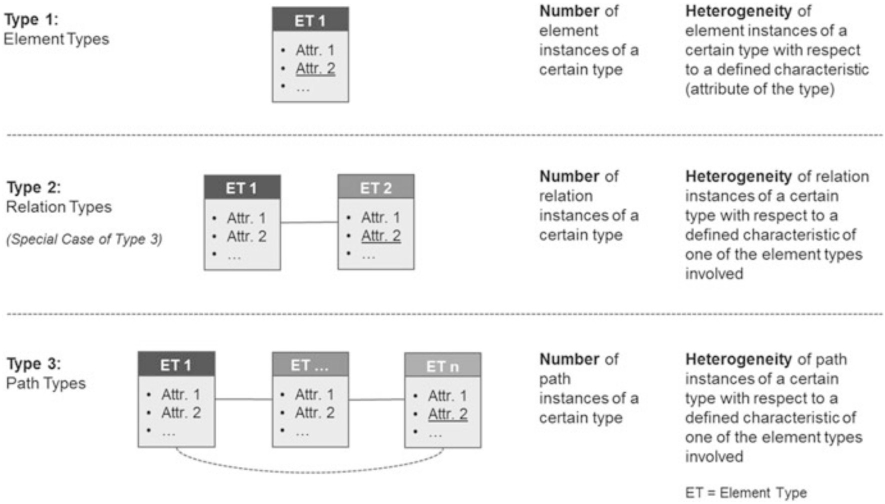

Fig. 13.7 Complexity aspects in metamodels (Schmidt 2013)

13.3.2 Extending the Basic Approach

While the existing heterogeneity-based approach has proven to be very versatile and powerful, it also has revealed some limitations (Schmidt 2013). In particular, it does not fully capture the internal *structure* of an architecture. The interfaces within an application landscape, for instance, may be analyzed for a concentration on applications or interface technologies. However, this does not account for the degree of *modularity* within the landscape. Yet, any experienced architect will agree that a modular application landscape divided into a set of loosely coupled application domains with interfaces predominantly within domains and only few interfaces crossing the domain boundaries will be much easier to manage than a landscape with interfaces placed at random. This is mainly due to a limitation in *change propagation* as the effects of changes can be contained within the respective domains. It may hence be argued that a complexity analysis should also address an evaluation of the architecture's modularity.

As shown by Aier and Schönherr (2007) and Simon and Fischbach (2013), architectural modularity may be assessed using the measure introduced by Newman (2006). For this purpose, the architecture (or parts hereof) needs to be transformed into a plain network consisting of nodes (e.g., applications) and edges (e.g., application interfaces). Modularity can then be calculated as the number of edges that fall into a set of given groups (modules, clusters) minus the expected number in an equivalent network with random edges (Newman 2006). It takes a value in the range between −0.5 and 1 with positive values indicating a *concentration* of edges within modules above the level expected for a random distribution.[6] Modularity can be determined with respect to a predefined clustering structure (e.g., an existing domain model). Alternatively, it may be looked for a previously unknown (inherent) clustering structure using dedicated search algorithms (see Newman 2006).

Another shortcoming of the basic approach is its focus on absolute figures. As shown in Sect. 13.1, the complexity of certain architectural domains or layers cannot be assessed in isolation. It rather needs to be put into relation with the requirements of superimposed domains. Therefore, *relative measures* capturing the quantity or variety of elements and relationships in relation to each other may be more appropriate. For example, the number of applications (from the application architecture) may be related to the number of business functions (in the business architecture). Similarly, the variety of platform technology may be put into relation with the variety of application systems.

Integrating these extensions with the basic approach results in an extended set of generic complexity measures that can be applied to arbitrary domains and at varying levels of detail. This is summarized in the *framework of complexity measures* depicted in Table 13.1. In the next section, the framework is applied to the domain of business architecture.

13.4 Measures for Business Architecture Complexity

Few authors have so far addressed the topic of business architecture complexity (e.g., Gottfredson and Aspinall 2005; Mocker and Ross 2013). Up to now, no systematic approach has been proposed to measure business complexity in its various aspects. Therefore, in this section, the generic framework presented in Sect.13.3 is applied to the *business architecture framework* introduced in Chap. 1. Starting with the business execution layer and progressing to business model and business motivation, the main concepts of business architecture are examined from a complexity perspective using the four complexity dimensions as a reference. Based on this, specific business complexity measures are proposed. This is illustrated by means of some examples.

13.4.1 Business Execution

According to Chap. 1, the business execution layer describes the business capabilities required by an organization and the way they are implemented in terms of processes, organization units, information objects, and so on. Obviously, complexity plays an important role in this area. Large organizations, for example,

[6] Referring to the four complexity dimensions of the basic approach as shown in Fig. 13.3, modularity may hence be interpreted as a special instance of relation variety (see Table 13.1).

Table 13.1 Framework of generic complexity measures

		Elements (E)	Relationships (R)
Quantity (Q)	Absolute	N_E: number of element instances (e.g., number of applications)	N_R: number of relation instances (e.g., number of business function implementations)
	Relative	$\frac{N_{E1}}{N_{E2}}$: number of element instances relative to each other (e.g., number of applications per number of business functions, irrespective of existing relationships)	$\frac{N_R}{N_E}$: number of relation instances relative to element instances (e.g., number of business function implementations per number of business functions)
Heterogeneity/ Variety/ Concentration (C)	Absolute	$EM_{E;A}$: entropy measure of element instances of type E by certain attribute A (e.g., concentration of applications by vendor)	$EM_{R;E;A}$: entropy measure of relation instances of type R (or path instances of type P) by certain attribute A of related element instances of type E (e.g., concentration of business function implementations by application names) $M_{R;E;A}$: modularity of relation instances of type R (or path instances of type P) with respect to attribute A of instances of type E (e.g., modularity of application interfaces along business domain names)
	Relative	$\frac{EM_{E1;A1}}{EM_{E2;A2}}$: entropy measure of element instances of type E_1 by attribute A_1 relative to entropy measure of element instances of type E_2 by attribute A_2 (e.g., concentration of technical platforms by vendors in relation to concentration of applications by vendors)	$\frac{M_{R1;E1;A1}}{M_{R2;E2;A2}}$: modularity of relation instances of type R_1 (or path instances of type P_1) with respect to attribute A_1 of instances of type E_1 relative to modularity of relation instances of type R_2 (or path instances of type P_2) with respect to attribute A_2 of instances of type E_2 (e.g., modularity of application interfaces in relation to modularity of business services)

often comprise hundreds of legal entities, processes, or sites, with major functional overlaps and redundancies. This type of complexity is well known in practice and various management methods like business process reengineering or lean management have been proposed do deal with it. But how can business execution complexity be formally described and measured?

Commencing with the *business capabilities*, a complexity assessment may start by looking at the *number* of (logical) capabilities in scope of the organization. Assuming that all capabilities are about equal in functional size, organizations with a larger number of capabilities may be considered more complex.[7] A fashion group, for instance, that maintains internal capabilities for the whole value chain from product design and marketing to manufacturing and sales, may be attributed a higher functional complexity than a competitor that is focusing on product design, marketing, and sales while relying on low-wage contractors for the manufacturing part.

In addition to that, the relationships between the business capabilities may be seen as important determinants of business execution complexity as well. In general, organizations with a higher *number* of *interdependencies* between capabilities (cf. Chap. 10) may be considered more complex than such with few relationships. As an example, a strongly integrated military forces organization comprising different highly interrelated capabilities like missile, missile-defense, and airborne surveillance may be attributed a higher functional complexity than a less integrated manufacturer of consumer goods.

Beyond that, complexity may be assessed along the *variety* or *concentration* of *dependencies* between capabilities. In particular, organizations with a higher degree of capability *reuse* (as expressed in a larger dependency concentration) may—ceteris paribus—be considered less complex than organizations with a lower degree of capability reuse. Similarly, capability networks with a higher level of *modularity* may be considered less complex as they will mitigate change propagation, making it easier to manage change and preserve organizational agility.

Even more than at the logical level, business execution complexity is determined at the physical level, i.e., the level of *capability realization*. It is here that duplication occurs and waste is created. In practice, most larger organizations comprise multiple (and at least partially redundant) implementations of the same capabilities. The procurement capability of a pharmaceutical group, for instance, may be implemented multiple times across different countries and deploying different variations of the same process type.[8]

Capability realization complexity can be captured in two ways. First, functional *redundancy* may be determined by relating the *number* of capability realizations or configurations[9] to the overall number of (logical) capabilities. The more such configurations exist per logical capability, the more duplication of work and the

[7] It should be noted that the actual complexity figures are strongly dependent on the used capability model and the associated level of detail. As a consequence, comparisons over time or between peers need to be based on the same reference model (or at least modeling guidelines) to be of any meaning.

[8] While there is good reason for the emergence of such architectures (e.g., historical evolution based on mergers and acquisitions), it is clearly in conflict with the goal of architectural efficiency as defined in Sect. 13.1.

[9] Capability configurations may be defined as existing combinations of business processes, organization units, information objects, resources, people, and culture in an organization realizing a certain business capability.

Table 13.2 Complexity measures for the business execution layer

Elements	Type		Measure	Interpretation
Business capability	E	Q	*Number* of (logical) business capabilities	Measures the "functional breadth" of the organization
	R	Q	*Number* of *relations* between (logical) business capabilities	Measures the degree of "functional interdependency" within the organization
		C	*Concentration* of capability dependencies by (re-) used capabilities	Measures the degree of "functional reuse" (logical level) within the organization
			Modularity of the business capability network	Measures the "functional decoupling" (logical level) within the organization (may serve as an indicator for change propagation and agility)
Capability realization (combination of process, organization unit, etc.)	E	Q	*Number* of business capability realizations per number of (logical) business capabilities	Measures the "functional redundancy" of the organization
	R	C	*Variety* of business capability implementations with respect to business processes, organization units, etc.	Measures the variety in capability implementation with respect to processes, organization units, etc.

more waste of resources will occur. Second, the *variety* of the implementation may be assessed with respect to processes, locations, etc. A manufacturing group, for example, whose capability implementations have been concentrated on 3 sites, may be considered less complex than a peer operating 30 sites across 20 countries (with differing legal frameworks, etc.). The same applies to the concentration of locations, processes, and so on.

The complexity aspects presented to assess the complexity of the business execution layer are summarized in Table 13.2. Their application is illustrated in Fig. 13.8.

The proposed measures are universal in the sense that they can be applied to different parts of an organization and at varying levels of detail. However, it should be noted that the optimal level of complexity may vary across different parts of the organization. Commodity services (like procurement, finance, IT, etc.), for example, may be assigned more ambitious complexity targets, because they can be standardized across business lines or regions. Capabilities required to differentiate in the marketplace, on the other hand, may make higher levels of complexity inevitable (e.g., to improve customer experience and satisfaction). To account for this, the complexity analysis needs to be extended to the overarching business model.

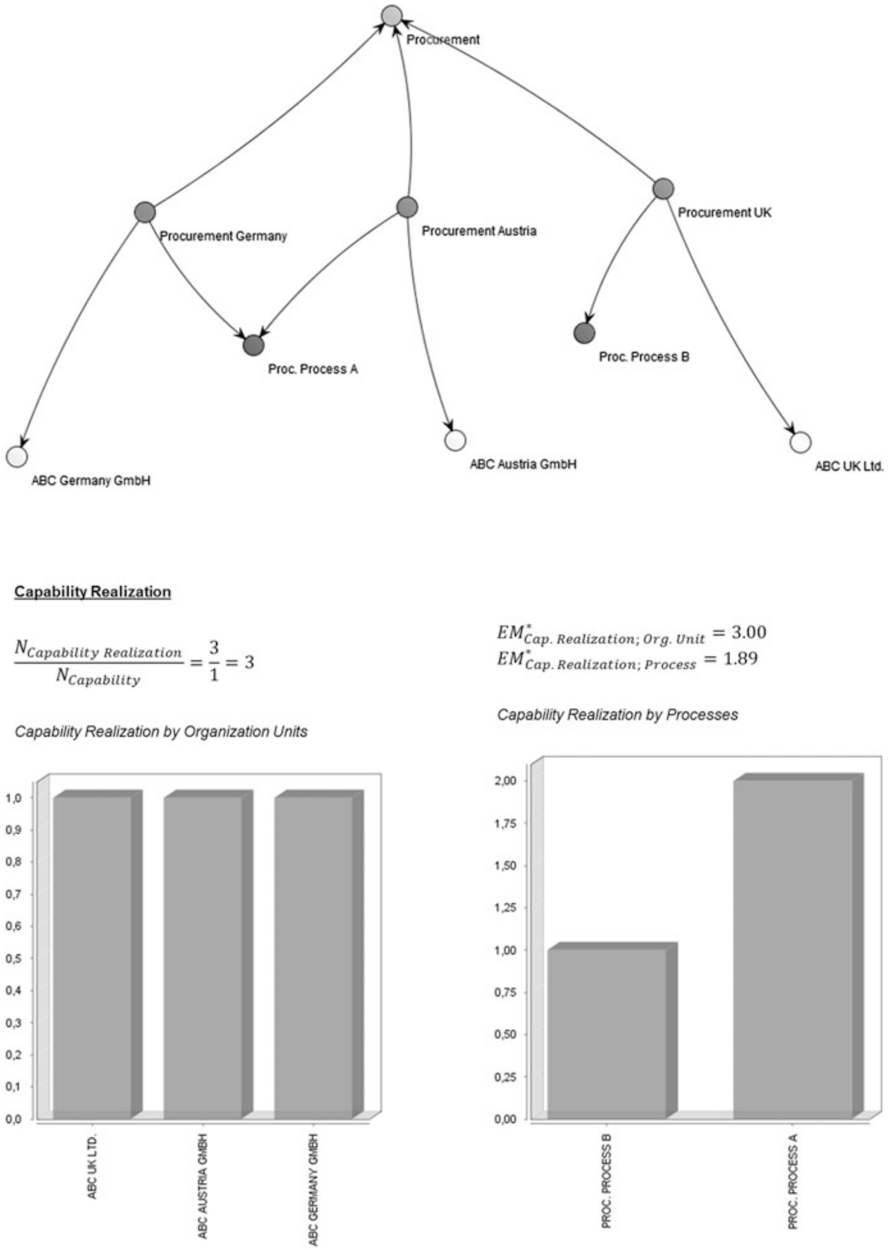

Fig. 13.8 Example for business execution complexity (procurement capability)

13.4.2 Business Model

While the concept of complexity is often used in relation to the business execution layer, it is less frequently applied to the business model.[10] Yet, it is the business model that sets the scene and the requirements for the business execution. An assessment of business complexity hence cannot be complete without a reference to the business model. What may be a perfect level of business execution complexity for an integrated technology group offering a variety of interrelated producer goods and related services (e.g., medical technology) may be completely inappropriate for a manufacturer of physical consumer goods (e.g., household appliances). But again, how can the complexity of the business model be formally described and measured?

First and foremost, business model complexity may be assessed along the *number* of core elements constituting the *network of value creation*. Depending on the industry, these elements will generally include customer segments, products/services, distribution channels, supplier segments and supplier channels as well as the key activities taken up in the overall value chain.[11] The more of these elements exist, the more complex the network of value creation will be—given that all other parameters are left unchanged. As an example, the business model of a universal bank that serves many different customer segments including retail, high networth individuals, small businesses, and large multinational corporations may be considered more complex than that of a private bank serving only a small segment of wealthy individuals. Similarly, the business model of a direct bank using the Internet and call centers as the only distribution channel may be considered less complex than that of a traditional commercial bank serving a larger number of distribution channels including physical branches, agents, call centers, and online channels. The same applies to the number of products and supplier segments.

Second, business model complexity is impacted by the *variety* of the core elements. An organization with a highly heterogeneous set of products and services like Samsung, for example, including mobile devices, household appliances, and power plants, may be considered more complex than a company like Apple with a very focused offering. The same applies to customer segments, distribution channels, key activities, and supplier segments.

In addition to the number and variety of core elements, the *number* of *relationships* between these are important determinants of the business model complexity as well. The more such relationships are in place, the more variants (or configurations) of value creation exist within the business model. Obviously, a bank that serves its retail clients only through online channels and its high-

[10] The few authors who have addressed this include Gottfredson and Aspinall (2005) and Mocker and Ross (2013).

[11] In addition to the core elements of the value creation network, complementing elements like revenue streams and pricing models, cost structures, value chain coordination mechanisms, or assets (see Chap. 1) could be analyzed in a similar way.

networth-individual clients exclusively through private banking branches (two relations) will have less variation in value creation than a bank that serves both customer groups through both channels (four relations).

Also, the *variety* of the relations may be assessed with respect to different dimensions. A business model with a higher level of concentration of the value creation configurations with regard to certain distribution channels or customer groups may be considered less complex than a business model that employs all these elements at equal weight.

Taking a closer look at the *product/service offering*, business model complexity may also arise from the dependencies between individual products/services. An organization with a large *number* of product/service *dependencies* may be said to be more complex from a product/service portfolio perspective than an organization with only few such dependencies. However, given a certain number of dependencies, complexity may be considered lower if products and services are based on a small set of reusable base products/services. In a commercial bank, for example, various different checking account products (e.g., with/without branch service, for students/adults, with/without savings account) may be based on a common base product. Therefore, the *concentration* of product/service *dependencies* may be an additional indicator for product/service complexity.

In larger organizations, *common business services* are often centralized within dedicated service units. These service units then act as internal service providers to a number of consumers (e.g., country-specific entities) within the group (*service-oriented architecture*). In such settings, business models can be described for each service provider. Beyond that, the interactions taking place between service providers and service consumers may be analyzed. This is of particular relevance from a group complexity perspective. For this purpose, the *structure* of service *dependencies* between legal entities/service units may be evaluated. Groups with a higher degree of service *reuse* (as reflected in a larger service usage concentration) may—ceteris paribus—be considered less complex than organizations with a lower degree of service reuse. Similarly, service networks with a higher level of *modularity* may be judged less complex, as they will mitigate change propagation and make it easier to manage change and preserve organizational agility.

The complexity measures presented to assess the complexity of the business model layer are summarized in Table 13.3. Their application is illustrated in Fig. 13.9.

The proposed measures can be applied to different parts of an organization and at varying levels of detail. Large organizations often employ different business models for their main business lines. Conglomerates like Siemens or General Electric, for example, may follow completely different business models for power, transportation, and health technology. In such organizations, the business model complexity may be assessed separately for each business line. In addition to that, the *number* of business models and the *number* of *relationships* between these may be regarded as further determinants of the overall business model complexity of the group.

Table 13.3 Complexity measures for the business model layer

Elements	Type		Measure	Interpretation
Customer segment	E	Q	*Number* of customer segments	Measures the "customer breadth" of the business model
		C	*Variety* of customer segments according to certain attributes (e.g., customer segment type)	Measures the heterogeneity of the customer segments
	R	Q	*Number* of *relations* between customer segments and products/services, distribution channels, supplier segments/ channels, key activities, etc.	Measures the dependency between the customer segment dimension and other dimensions of the business model
		C	*Concentration* of customer segments along products/ services, distribution channels, supplier segments/channels, key activities, etc.	Measures the "customer segment concentration" along other dimensions of the business model
Product/ service	E	Q	*Number* of products/services	Measures the "breadth" of the product/service portfolio
		C	*Variety* of products/services according to certain attributes (e.g., product class)	Measures the heterogeneity of the product/service portfolio
	R	Q	*Number* of *relations* between products/services and customer segments, distribution channels, supplier segments/channels, key activities, etc.	Measures the dependency between the product/service dimension and other dimensions of the business model
		C	*Concentration* of products/ services along customer segments, distribution channels, supplier segments/channels, key activities, etc.	Measures the "product/service concentration" along other dimensions of the business model
		Q	*Number* of dependencies between products/services	Measures the degree of "product/ service interdependency" of the business model
		C	*Concentration* of product/ service dependencies	Measures the degree of "product/ service reuse"
			Modularity of product/service dependencies	Measures the degree of "product/ service decoupling"

(continued)

Table 13.3 (continued)

Elements	Type		Measure	Interpretation
Distribution channel	E	Q	*Number* of distribution channels	Measures the "distribution channel breadth" of the business model
		C	*Variety* of distribution channels according to certain attributes (e.g., channel type)	Measures the "distribution channel variety"
	R	Q	*Number* of *relations* between distribution channels and customer segments, products/ services, supplier segments/ channels, key activities, etc.	Measures the dependency between the distribution channels and other dimensions of the business model
		C	*Concentration* of distribution channels along customer segments, products/services, supplier segments/channels, key activities, etc.	Measures the "distribution channel concentration" along other dimensions of the business model
Supplier segment	E	Q	*Number* of supplier segments	Measures the "supplier breadth" of the business model
		C	*Variety* of supplier segments according to certain attributes (e.g., supplier segment type)	Measures the "supplier segment variety"
	R	Q	*Number* of *relations* between supplier segments and customer segments, products/services, distribution channels, key activities, etc.	Measures the dependency between supplier segments and other dimensions of the business model
		C	*Concentration* of supplier segments along customer segments, products/services, distribution channels, key activities, etc.	Measures the "supplier segment concentration" along other dimensions of the business model
Key activity	E	Q	*Number* of key activities	Measures the "activity breadth" of the business model
		C	*Variety* of key activities according to certain attributes (e.g., activity type)	Measures the variety of key activities
	R	Q	*Number* of *relations* between key activities and customer segments, products/services, distribution channels, supplier segments, etc.	Measures the dependency between key activities and other dimensions of the business model
		C	*Concentration* of key activities along customer segments, products/services, distribution channels, supplier segments, etc.	Measures the concentration of key activities along other dimensions of the business model

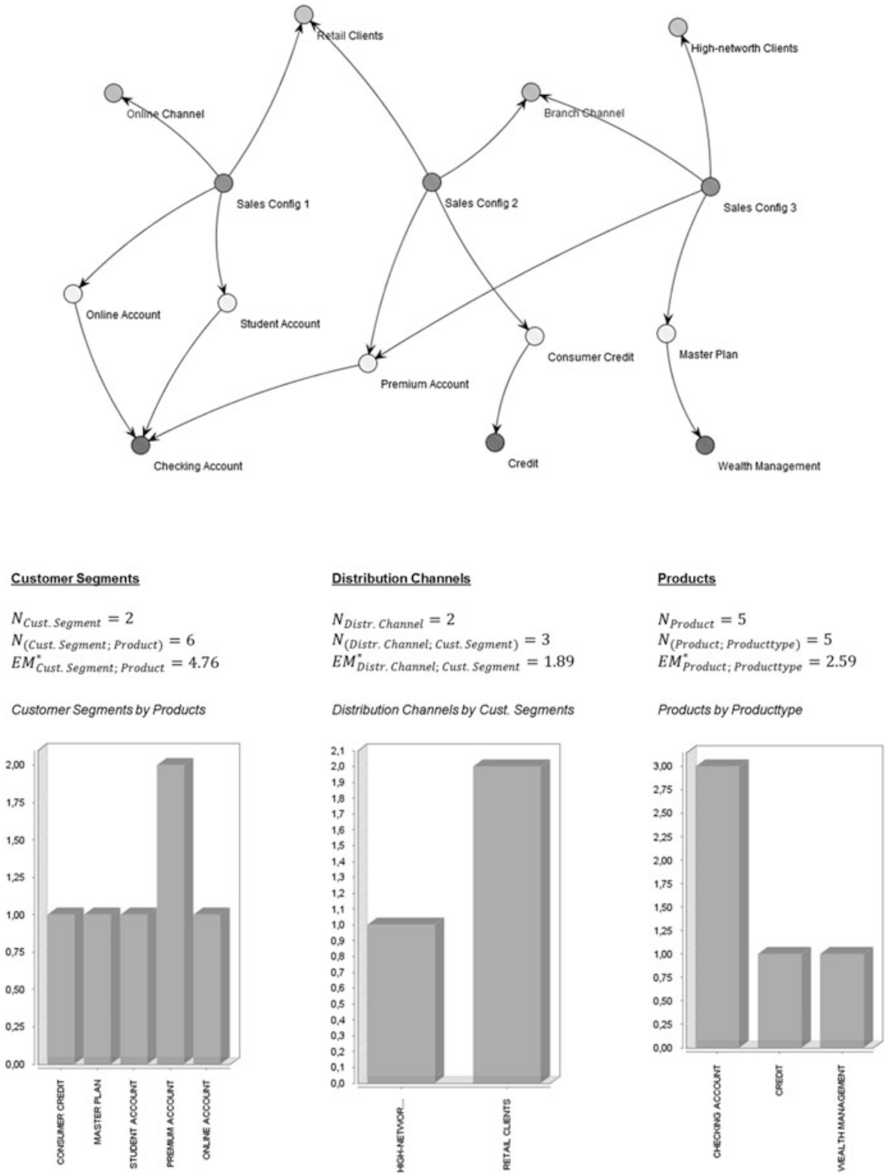

Fig. 13.9 Example for business model complexity (commercial bank)

As with the business execution, the optimal level of complexity may vary depending on the complexity of the business environment and the goals and objectives of the organization. A strongly regulated market environment like the pharmaceutical industry, for example, may impose special requirements to the business model (e.g., distribution to the end consumer via licenced pharmacies based on medical prescription only), hence setting limits to the minimum complexity possible. Therefore, an assessment of the business model complexity needs to take into account the underlying business motivation.

13.4.3 Business Motivation

As shown in the previous sections, the concept of complexity applies well to the domains of business execution and business model. But how about the business motivation? Clearly, the business model followed by an organization should be in line with the overarching system of environmental/boundary conditions and organizational objectives.[12] To achieve an optimal alignment, all these factors must be addressed while maintaining a minimal level of complexity. An insurance company, for example, with ambitious profitability objectives, operating in a strongly regulated and highly competitive environment, may need to adopt a more sophisticated business model that leverages the expertise of independent agencies to grow into a set of profitable niche markets. The same business model may be far too complex (and risky) for an organization operating in a much more simple business environment. But how can the complexity of the business motivation be formally described and measured?

According to Chap. 1, the business motivation layer is comprised of business influencers (internal and external drivers and constraints), business ends (esp. mission, goals, objectives), and business means (strategies, business directives/principles). Just like with the business execution and business model layer, these may be described as a network of elements and relationships. A strategy, for example, may be related to a number of objectives that it is supporting. The objectives may, in turn, be linked to the business goals that they are based on. Finally, the business goals may be associated with certain internal and external drivers. Based on such a network of business influencers, business ends, and business means, complexity may be analyzed as follows.

At the top level, business motivation complexity may be assessed along the *number* of *business influencers*. Depending on the industry, business influencers will generally include shareholder requirements (e.g., regarding minimum dividends), regulatory requirements (e.g., applicable laws and accounting standards), market conditions (e.g., degree of competition), technological developments (e.g., new materials or production methods), and so on. The more such drivers and/or constraints exist, the more complex the business motivation may be considered. A bank operating under the Basel III regime, for example, will have to comply with a larger number of (frequently changing) regulatory constraints than a retailer for consumer electronics. Where business influencers can be categorized into certain classes, the *variety* may be taken into account as well.

[12] This corresponds with the EA school of “Enterprise Ecological Adaption” as introduced by Lapalme (2012).

At the second level, business motivation complexity may be assessed along the *number* of *business ends*. These generally include the mission, goals, and objectives followed by the organization. The more of these exist, the more complex the business motivation may be considered.[13] A utility firm, for example, may notice an increase in motivational complexity if environmental goals (e.g., CO^2 reduction) are added to financial and organizational goals. If business means are categorized into certain dimensions (e.g., using a "Balanced Scorecard"), the *variety* may be analyzed as well.

In addition to that, business motivation complexity may be assessed along the *number* of *dependencies* between business ends, but also with respect to business influencers. The more such dependencies are in place, the more complex the business motivation may be considered.

Also, the *variety* of the *relations* may be evaluated. A business motivation with a higher level of concentration of dependencies on a small set of common goals and objectives may be considered less complex than a business motivation that connects to all goals and objectives in an equal way.

Finally, business motivation complexity may be analyzed along the *number* of *business means*. These typically include strategies and directives. The more of these elements, the more complex the business motivation may be considered. A reinsurer, for example, that follows different strategies for the "Life & Health" and the "Property & Casualty" business may be considered more complex from a strategic perspective than a competitor that attempts to address these lines of businesses with the same general strategy.[14]

Beyond that, the *number* of *relationships* between the individual business means need to be taken into account. The more such dependencies are in place, the more complex the business motivation may be considered. Google, for example, with its large number of interrelated strategies for different segments from online advertising, mobile device ecosystems to household appliances—each one serving the others—may be attributed a higher level of strategic complexity than a traditional manufacturer of mobile devices. Again, the *variety* of the *relations* may be assessed. A business motivation with a higher level of concentration of dependencies on a small set of strategies and directives may be considered less complex than a business motivation where all business means are of equal importance.

The complexity measures presented to assess the complexity of the business model layer are summarized in Table 13.4. Their application is illustrated in Fig. 13.10.

[13] Like with other element types, measures will be strongly dependent on the actual modeling of goals and the chosen level of detail. As demonstrated in Chap. 2, goals should be defined in an atomic way with each goal addressing only one aspect.

[14] However, such a differentiation may be a strategic necessity given varying market conditions.

Table 13.4 Complexity measures for the business motivation layer

Elements	Type		Measure	Interpretation
Business influencer (driver, constraint)	E	Q	*Number* of business influencers	Measures the degree of internal and external influence in the business motivation
		C	*Variety* of business influencers according to certain attributes (e.g., internal, regulatory)	Measures the variety of the internal and external influence
Business end (mission, goal, objective)	E	Q	*Number* of business ends	Measures the "size" of the organizations' system of business ends
		C	*Variety* of business ends according to certain attributes (e.g., Balanced Scorecard)	Measures the variety of business ends by type
	R	Q	*Number* of dependencies between business ends and business influencers	Measures the degree of dependency between business influencers and business ends
		C	*Variety* of dependencies between business ends and business influencers	Measures the concentration of business influencers along business ends
		Q	*Number* of dependencies between business ends	Measures the degree of dependency between business ends
		C	*Variety* of dependencies between business ends	Measures the concentration of relationships between business ends
Business means (strategy, directive, principle)	E	Q	*Number* of business means	Measures the range of business means
	R	Q	*Number* of *dependencies* between business means (e.g., strategies affecting each other)	Measures the degree of strategic dependency
		C	*Variety* of *dependencies* between business ends and business means	Measures the concentration of business ends on business means

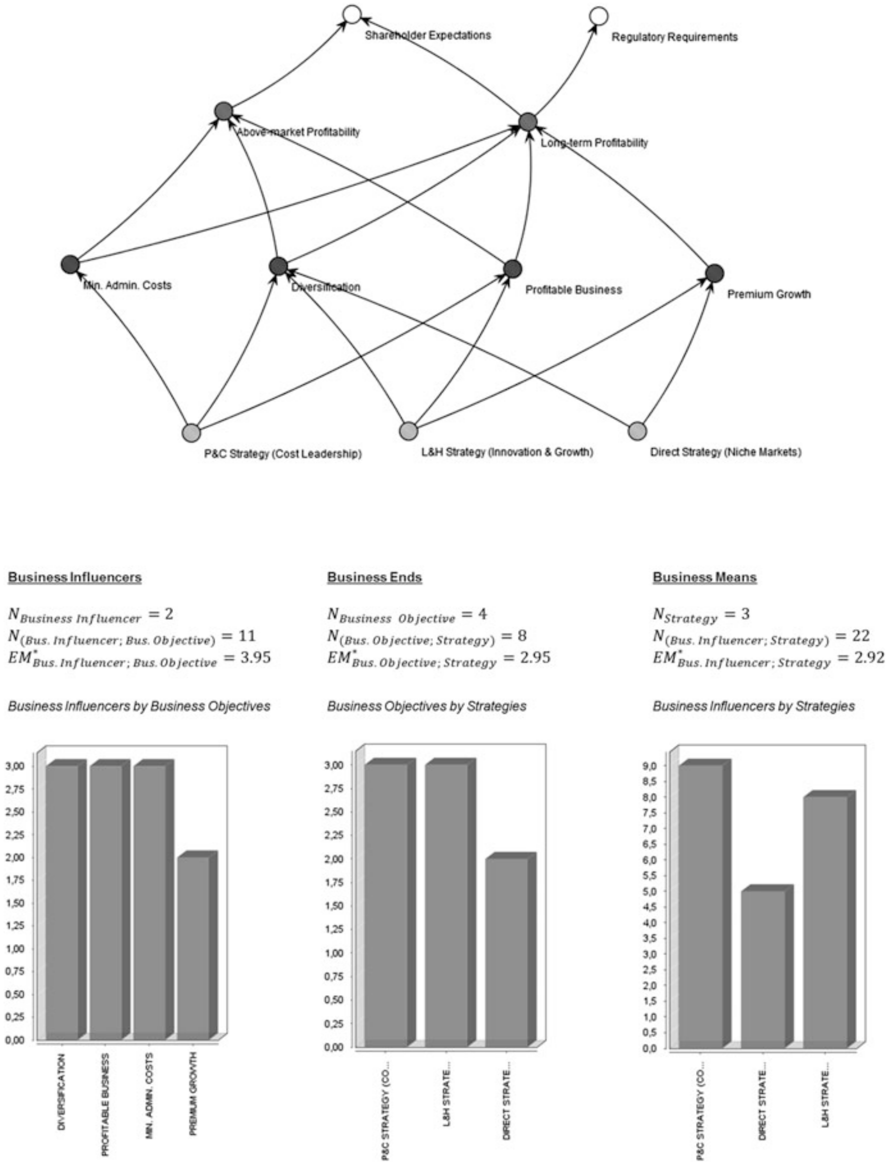

Fig. 13.10 Example for business motivation complexity (reinsurer)

13.5 Putting Business Complexity Measures to Practice

In the previous section, a broad range of possible measures for assessing the complexity of business architectures has been presented. To put these into practice successfully, some additional steps need to be taken. Most importantly, the concept of complexity measures needs to be introduced to the target organization and the required measures and reports need to be implemented.

13.5.1 Introducing Complexity Measures

Implementing a complexity reporting in a given organization may be a difficult task. In order to create maximum value for the organization and its stakeholders, the particular context and requirements of the organization need to be taken into account. This is of special importance, as implementing a complexity reporting will generally lead to additional efforts in the short run. Beyond that, there is a risk that complexity measures are misunderstood or misused. By some stakeholders, they may even be perceived as a threat. For these reasons, and in line with the method proposed in Chap. 15, the introduction of a business complexity reporting should be based on a thorough analysis of stakeholder needs. During such an analysis, the main application scenarios should be reviewed and prioritized.[15] In general, complexity measures may be employed for the scenario types shown in Table 13.5.

In addition to these general scenarios, complexity management may be focused on certain architecture layers or domains depending on the given situation and context of the organization (cf. Chap. 15). For example, a company with a well-defined and lean business model but a large extent of redundancy in operations may

Table 13.5 Application scenario types

Application scenario	Description
Decision support/ simulation	Calculation (and comparison) of complexity impacts for major decision variants (e.g., target architecture scenarios)
Comparative analysis	Calculation (and comparison) of complexity measures for different parts of the overall landscape (e.g., between domains, business lines, etc.) and identification of hot spots
Benchmarking	Calculation (and comparison) of complexity measures for different organizations in search for best-in-class complexity figures
Architectural controlling	Systematic planning of target complexity figures as part of a continuous architecture management (e.g., differentiated by architecture layers and domains)
Risk management	Calculation of complexity measures as part of the internal management of operational risks

[15] This may be strongly facilitated by using a detailed catalog of typical application scenarios as a reference.

concentrate on business execution complexity in the first place. Similarly, the scope may also be set to certain domains that need special attention (e.g., harmonization of procurement, accounting, etc.). Based on such a priorization, the measures of primary relevance may be selected for implementation.

13.5.2 Technical Implementation

After the application scenarios and required measures have been identified, methods must be implemented to calculate the actual figures based on available data and to create appropriate reports. Generally, the calculation of measures should be based on the data captured in the architecture repository. This way, the calculation of measures can be automated and fully integrated with existing architecture data management processes. Data elements that are not yet available in the repository should be added first. The calculation of measures may then either be implemented within the architecture management tool or based on specialized tool support for complexity management.[16]

13.6 Conclusion and Outlook

In this chapter, a generic approach for measuring complexity was presented and applied to the domain of business architecture. It was shown that the notion of complexity is of relevance not only to the business execution but also to the business model and business motivation layer. A more complex system of business drivers, goals, and objectives is more likely to require a more complex business model. This is turn will generally call for a more sophisticated (and thus complex) business execution layer. For all these layers, a number of measures were proposed and illustrated by examples. Using these measures may be supportive in minimizing the complexity surplus and optimizing the overall architectural alignment.

However, assessing and managing business architecture complexity remains a difficult task. First of all, broad stakeholder buy-in is required in order to gain visibility and acceptance for such an initiative. Second, data needs to be captured and maintained in an appropriate form. This will generally require an extension of existing architecture repositories and the respective data maintenance processes. Also, measures need to be adjusted to the organization-specific metamodel. As the actual figures are strongly dependent on the modeling approach and associated level of detail, care must be taken to ensure that appropriate reference models and modeling guidelines are used consistently. Beyond that, appropriate methods need to be defined to handle missing data elements. Last but not least, it must be

[16] An example for such a specialized tool kit is the Plexity Analyzer™, which supports a flexible configuration and calculation of all relevant complexity measures based on arbitrary data/metamodel structures.

emphasized that complexity assessments often require thorough analyses that can only by carried out by skilled and experienced architects. Such analyses will comprise drill-down operations and supplemental research. The results should hence always be commented and interpreted qualitatively.

Further research is required to determine the relevance of the proposed measures in more detail and to evaluate their practical use. Beyond that, methods for aggregating the complexity measures and relating the figures of different layers to each other need to be developed. Generally, more research is required on how to actually manage complexity given that appropriate complexity measures are in place. As initial results from the IT architecture domain indicate, complexity is not a one-dimensional variable that can easily be reduced across all its facettes (Schmidt 2013; Schmidt et al. 2013). Instead, it appears that reducing the complexity of a certain aspect (e.g., number of applications, vendor variety) will often lead to an increase of complexity in another aspect (e.g., number of application usage). Also, the impact may vary between a local and a global perspective. Application consolidation, for example, will typically lead to a reduced number of applications and a reduced vendor variety. However, the dependencies of the particular target application will generally rise (in terms of interfaces with other system, usage relations, country/languages, etc.). From a local perspective, the complexity will hence increase. It may be concluded that similar mechanisms apply to the consolidation and centralization of business capabilities or the streamlining of business models, for example. Additional research is required to evaluate this in more detail and to give business architects appropriate methods at hand.[17]

References

Aier S, Schönherr M (2007) Integrating an enterprise architecture using domain clustering. In: Lankhorst M, Johnson P (eds) Proceedings of the Second Workshop on Trends in Enterprise Architecture Research (TEAR 2007), St. Gallen, Switzerland, 6 June 2007, pp 23–30

Drucker PF (1974) Management: tasks, responsibilities, practices. Harper & Row, New York

Gottfredson M, Aspinall K (2005) Innovation versus complexity. What is too much of a good thing? Harv Bus Rev 83(11):62–71

Lagerström R, Baldwin CY, Maccormack AD, Aier S (2013) Visualizing and measuring enterprise application architecture: an exploratory telecom case. School working paper 13-103. Harvard Business, Cambridge, MA

Lapalme J (2012) Three schools of thought on enterprise architecture. ITPro, November/December 2012, IEEE Computer Society, pp 37–43

Lehmann MM (1997) Laws of software evolution revisited. In: Montangero C (ed) Software process technology—proceedings of the 5th European workshop on software process technology, Nancy, Oct 1996 (Lecture notes in computer science), vol 1149. Springer, Berlin, pp 108–124

[17] In addition to the framework presented in this chapter, centrality measures from the domain of network analysis may be used to assess local complexity and counteract global "over-optimization" (cf. Simon and Fischbach 2013).

Mocker M (2009) What is complex about 273 applications? Untangling application architecture complexity in a case of European investment banking. In: Proceedings of the 42nd Hawaii international conference on system sciences (HICSS), IEEE Computer Society, Washington, DC

Mocker M, Ross JW (2012) USAA: capturing value from complexity. CISR working paper No. 389, MIT Sloan School of Management, Cambridge, MA

Mocker M, Ross JW (2013) Rethinking business complexity. CISR Research Briefing XIII(2), MIT Sloan School of Management, Cambridge, MA

Newman MEJ (2006) Modularity and community structure in networks. Proc Natl Acad Sci USA 103(23):8577–8582

Rutz U (2012) IT complexity management @ Commerzbank—overview on Commerzbank initiative. Presentation at EAMKON conference, Stuttgart, Apr 2012

Schmidt C (2013) How to measure enterprise architecture complexity: a generic approach, practical applications and lessons learned. Presentation at The Open Group conference, London, Oct 2013. www.scape-consulting.com/index.php/schmidt-2013d.html. Accessed 25 Oct 2014

Schmidt C, Buxmann P (2011) Outcomes and success factors of enterprise IT architecture management: empirical insight from the international financial services industry. Eur J Inf Syst 20(2):168–185

Schmidt C, Widjaja T, Schütz A (2013) Messung der Komplexität von IT-Landschaften auf der Basis von Architektur-Metamodellen: Ein generischer Ansatz und dessen Anwendung im Rahmen der Architektur-Transformation. In: Horbach M (ed) INFORMATIK 2013, GI-Edition (Lecture notes in informatics (LNI) P-220). Köllen Verlag, Bonn, pp 1261–1275

Schneberger SL, McLean ER (2003) The complexity cross: implications for practice. Commun ACM 46(9):216–225

Schneider AW, Zec M, Matthes F (2014a) Adopting notions of complexity for enterprise architecture management. In: Proceedings of the 20th Americas conference on information systems (AMCIS), Savannah, GA

Schneider AW, Reschenhofer T, Schütz A, Matthes F (2014b) Empirical results for application landscape complexity measures. In: Proceedings of the 48th Hawaii international conference on system science (HICSS), Kauai, HI

Schütz A, Widjaja T, Kaiser J (2013) Complexity in enterprise architectures—conceptualization and introduction of a measure from a system theoretic perspective. In: Proceedings of the 21st European conference on information systems (ECIS), Utrecht, The Netherlands

Shannon CE (1948) A mathematical theory of communication. Bell Syst Tech J 27:379–423, pp 623–656

Simon D, Fischbach K (2013) IT landscape management using network analysis. In: Poels G (ed) Enterprise information systems of the future, CONFENIS 2012, Ghent, Belgium. LNBIP, vol 139. Springer, Berlin, pp 18–34

The Open Group (2013) ArchiMate 2.1 specification. Van Haren Publishing, Zaltbommel

Widjaja T, Kaiser J, Tepel D, Buxmann P (2012) Heterogeneity in IT landscapes and monopoly power of firms: a model to quantify heterogeneity. In: Proceedings of the 33rd international conference on information systems (ICIS), Orlando, FL