

This article was downloaded by: 10.2.97.136

On: 31 Mar 2023

Access details: *subscription number*

Publisher: *Routledge*

Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: 5 Howick Place, London SW1P 1WG, UK



## **The Routledge Companion to Managing Digital Outsourcing**

Erik Beulen, Pieter M. Ribbers

### **Enterprise architecture**

Publication details

<https://test.routledgehandbooks.com/doi/10.4324/9781351037785-4>

Marlies van Steenbergen

**Published online on: 28 Jul 2020**

**How to cite :-** Marlies van Steenbergen. 28 Jul 2020, *Enterprise architecture from: The Routledge Companion to Managing Digital Outsourcing* Routledge

Accessed on: 31 Mar 2023

<https://test.routledgehandbooks.com/doi/10.4324/9781351037785-4>

**PLEASE SCROLL DOWN FOR DOCUMENT**

Full terms and conditions of use: <https://test.routledgehandbooks.com/legal-notices/terms>

This Document PDF may be used for research, teaching and private study purposes. Any substantial or systematic reproductions, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The publisher shall not be liable for an loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

# 3

## ENTERPRISE ARCHITECTURE

*Marlies van Steenberg*

### 3.1 Introduction

To keep up with the rapid changes happening in technology, markets and society, enterprises must be flexible and adaptive. This flexibility must be designed into the structure of the enterprise. In accordance with The Open Group, we consider an enterprise any organization of people, processes and means that share a common goal [1]. Thus, an enterprise can be a company or institution, but it can also be part of a company or a network of cooperating parties. An enterprise is a complex system, acting in a complex environment with many types of stakeholders such as customers, shareholders, partners, suppliers, jobseekers and regulators.

Enterprise architecture is the discipline that concerns itself with designing the structure of the enterprise. The aim of enterprise architecture is to translate the ambitions and strategy of an enterprise into guidelines on how to structure the necessary processes, information systems and technology, to enable the continued realization of the strategy. It takes a holistic view of the enterprise, looking for a seamless integration of products and services, processes, information flows and technology.

A widely adopted definition of architecture is the ISO/IEC 42010:2007 definition, which defines architecture as “the fundamental organization of a system embodied in its components, their relationships to each other and the environment, and the principles guiding its design and evolution.” This definition indicates that enterprise architecture is about components, relationships and principles. In the case of enterprise architecture, the system is the entire enterprise. The components, an enterprise is made up of, are of various types, such as organizational components (departments, roles), processes, data and information, and technology (software, hardware). In this definition, enterprise architecture is regarded as an artifact, i.e. the models and principles defining the enterprise. When we talk about the development and application of the models and principles, we speak of enterprise architecture management.

#### 3.1.1 Enterprise architecture artifacts

An enterprise architecture consists of a descriptive part, depicting the various components and how they relate to each other and the environment, as well as a normative part, the

principles guiding the design and evolution of the enterprise structure. The descriptive part is represented in the form of models. These models can be of various levels of detail and they can focus on different parts of the enterprise. It is common to find a high-level enterprise-wide model spanning the entire enterprise, complemented with more detailed models of narrower scope, for instance a business domain (the production function) or an aspect (data). Models are time-bound in the sense that they depict a situation at a certain timespan in the past, present (current architecture) and/or future (target architecture). The normative part is expressed in the form of statements that constrain the available choices for the target models. These statements, too, can differ in level of detail and scope. High-level principles express what is to be achieved, specific rules express how things are to be achieved. The level of detail greatly impacts the level of freedom left to the designer.

Principles and target models represent choices in how an enterprise expects to best be able to achieve its ambitions and goals. These choices are not the same for all enterprises, for what constitutes a good choice depends on the situation. Both the ambitions of an enterprise and the context it finds itself in impact the content of the enterprise architecture. An enterprise in a dynamic environment requires different architectural choices than an enterprise in a stable environment. Also, an enterprise focused on customer intimacy requires different architectural choices than an enterprise focused on operational excellence. The enterprise architect faces the challenge to balance all factors and translate them into a solution that best meets the entirety of different stakeholder concerns and contextual constraints. In doing so, the architect balances requirements such as reliability, flexibility and efficiency.

### **3.1.2 Enterprise architecture management**

An enterprise architecture must be designed, maintained and applied. This is the responsibility of the enterprise architecture management function. The enterprise architecture management function takes the business strategy of the enterprise as its starting point and translates it into principles and models that fit this strategy, taking stakeholders and context into account. This implies that changes in the strategy may lead to changes in the enterprise architecture. Though the other way around is also possible: an enterprise architecture may also enable and stimulate changes in strategy [2]. The choices made in the enterprise architecture define the strategic room for maneuver. Investment in the enterprise architecture is an investment in the future ability to grasp opportunities and respond to social and market trends. Besides designing and maintaining the enterprise architecture models and principles, the enterprise architecture management function entails applying these models and principles in the development processes [3]. This is closely related to the governance processes of the organization. Decisions relating to the structuring of the enterprise's processes, information and technology are constantly made at the strategic, tactical or operational level. At each of these levels, it is important to ensure that no decisions are made that are contrary to the enterprise architecture. This means that decision-makers must have access to the architectural norms. This access must be organized, either by providing a role for architects in the decision-making process or by providing architectural knowledge to the decision-makers. What choices are made in this regard depends on the complexity of the situation.

Enterprise architecture management is essential to the execution of business strategy and as such it is a core competence of the enterprise. We use competence here in the sense of the ability to successfully perform certain actions and achieve certain results [4]. Examples of competences are business strategy, systems and process innovation, managing change and applications development. This is not to be confused with the concept of capability. When

enterprises turn competences into business value, we speak of a capability. A capability is the ability to achieve sustained superior performance and business value. An example of a capability is the IS capability, which is the ability of an organization to deliver business value from investments in IS/IT continuously. Competences emerge out of the integration and coordination of resources. A resource is a means an enterprise has at its disposal. Resources can be tangible or intangible. Examples of resources are buildings, computers, skills, knowledge, processes, brands and customer relationships.

Summarizing, the aim of enterprise architecture is to provide structure to an enterprise that fits its strategic ambitions. It does so from a holistic perspective, aiming for the seamless connection of products and services, processes, information flows and technology. The topic of enterprise architecture concerns both the enterprise architecture artifacts, to be distinguished in models and principles, and the enterprise architecture management function of developing and applying the enterprise architecture artifacts.

### 3.3 Architecture frameworks

Over the decades, the views on position and goal of enterprise architecture have evolved. In the 1980s, architecture was regarded as part of information planning. An information plan often had a scope of up to five years. The architecture had the form of a blueprint, was all-compassing and based on the idea that the world is malleable. In the 90s, IT architecture is no longer considered part of an information plan, but as a set of principles and models that exist in their own right and must be maintained. The aim of the IT architecture was mainly standardization in IT. Attempts were made to address organizations' wishes for more rapid change, but that appeared difficult. Around 2000 realization dawns that architecture should be not only about IT, but also about processes and products and services: business architecture is born [5]. Models are made distinguishing front, mid and back office, and customers, channels and products, organizations move from product-oriented to process-oriented, etc. Gradually, it becomes clear that business and IT architectures should be integrated in an enterprise architecture covering all aspects of the organization: products/services, processes, organizational structure, information flows, applications, middleware, hardware and network technology must be considered together. One cannot be seen separate from the other.

Over the decades, various architecture frameworks have been developed to guide the development and application of enterprise architecture [6]. These frameworks support the architects in doing their job. They provide structure as well as a common language between architects. Architecture frameworks differ in focus and in their perception of architecture, but most frameworks have in common that they make the connection between strategy and solutions and distinguish various perspectives. Most frameworks distinguish in some way or another between the perspectives of business, information systems and technical infrastructure. Dependent on the focus of the framework, attention may be paid to the architecture artifacts, the architecture processes and/or the organization of the architecture management function.

One of the first architecture frameworks was the Zachmann Framework [7]. The Zachmann Framework focuses on the enterprise architecture artifacts. It provides a logical structure for classifying and organizing the descriptive representations of an enterprise, covering the entire set of descriptions necessary to describe an enterprise, or any other system. The Zachmann Framework is a two-dimensional matrix. The horizontal axis is based on the interrogatives What, How, When, Who, Where and Why. The vertical axis is based on the transformation from an abstract idea into instantiation: Identification, Definition,

Representation, Specification, Configuration and Instantiation, which represents the perspectives of different stakeholders. This provides a matrix in which the 30 cells represent the classification of the framework. The framework purports to be normalized, complete and stable. The normalized primitives (the cells) can be combined into composites needed to “build” a working enterprise. The framework enables focusing on one component, without losing track of the complete picture and the context in which the component exists. The Zachmann Framework is considered by many the mother of all EA classifying frameworks. Besides the Zachmann Framework, various other similar frameworks have been developed, sometimes for specific domains. In practice today, Zachmann is more of a reference framework than an integrated framework that is implemented.

In the early 90s, TOGAF™, which stands for The Open Group Architecture Framework, was launched. TOGAF is developed and maintained by The Open Group Architecture Forum [8]. The first version of TOGAF was developed in 1995. The core of TOGAF is the Architecture Development Method (ADM), which describes the various phases in architecture development. It starts with a preliminary phase, in which the foundation is laid by tuning the framework to the organization and defining the foundational principles for architecture. After this groundwork, a cycle of eight phases is executed for each change initiative, the first phase being the definition of an architecture vision and the last phase the execution of architecture change management. TOGAF distinguishes four types of architecture as part of an enterprise architecture: business architecture, data architecture and application architecture (together referred to as Information Systems Architecture in the ADM) and technology architecture. The ADM cycle can be applied at various levels, such as the enterprise level, domain level or program level. It always starts with a need for a (business) change. Besides the ADM, TOGAF provides a lot of practices for enterprise architects to use, including guidelines and techniques to support the application of the ADM, the Architecture Content Framework that provides a detailed model of architectural work products, the Enterprise Continuum that provides a model for structuring a virtual repository and classification of architecture and solution artifacts, the TOGAF reference models: Technical Reference Model and Integrated Information Infrastructure Model, and the Architecture Capability Framework to help establish an architecture practice within an enterprise. TOGAF is an extensive framework with many useful practices from which an enterprise must make its own selection. The ADM provides a clear structure to the architectural processes.

ArchiMate is a modeling language. It was developed between 2002 and 2004 for architecture modeling and aims to contain all concepts that are relevant to describe an enterprise architecture [9]. The core concepts of the language are classified into aspects and layers. A concept may be labeled an active structure element (e.g. business actors or application components), a behavior element (e.g. business process or application function) or a passive structure element (e.g. business object or data object). A concept also belongs to one of three layers: business layer (e.g. business actors or business processes), application layer (e.g. application components or data objects) and technology layer (e.g. infrastructure function or node). The most recent version saw an extension of two additional layers: strategy and physical. Besides the core concepts, ArchiMate distinguishes concepts related to motivation, such as driver, goals and principle, and to migration, such as plateau and deliverable. A very important concept in ArchiMate is the service. The service is the externally visible behavior of a system. It is also the glue between the layers. Thus, the infrastructure layer is accessed by the application layer via infrastructure services. Similarly, the business layer can use the functions of the application layer via application services. The business layer delivers its services to the environment via business services. In 2008, ArchiMate was transferred to

The Open Group. ArchiMate is an architect's tool. It is very useful for accurate modeling as well as for communication among architects and designers. It is less suitable as a means of communication toward management. ArchiMate is supported by many modeling tools, including freeware.

In the 80s and 90s, the primary focus of most architecture frameworks was on the architecture artifacts. As a reaction to this, in 2002 Dynamic Architecture (DYA) was launched [10]. DYA focuses on the architecture process and on how to effectively apply architecture to realize strategic goals. It distinguishes four main processes: strategic dialogue, architecture services, development with architecture and development without architecture. The leading principles of DYA are a just enough, just in time approach, i.e. only developing architectural artifacts where and when there is a need, and the embedding of architecture in the change processes of the enterprise. DYA contains various instruments, such as the project-start architecture, an architecture artifact that tunes the enterprise architecture to the specific context of a project.

In more recent years, new frameworks were introduced, such as General Enterprise Architecting (GEA) [11] and Risk- and Cost-Driven Architecture (RCDA) [12].

Architecture frameworks are designed from the perspective of the architect. However, architecture is not a goal, but a means to achieve the business goals. To this end, architecture must be embedded in the change processes of the organization. These change processes recently underwent considerable changes with the rise of Agile development. With Agile, the focus shifted from central governance to autonomous development teams, with little or no room for architecture. Many of the premises of the existing architecture frameworks were no longer valid and architects struggled to establish new ways of working, adapting to the new development paradigms. Gradually, however, a new way of collaboration between architects and agile teams is emerging.

One Agile development framework that positions architecture within Agile development is SAFe®. SAFe® stands for Scaled Agile Framework [13]. It is developed and maintained by Scaled Agile, Inc. It is a scalable and configurable framework based on Lean-Agile principles and values. It helps organizations “deliver new products, services, and solutions in the shortest sustainable lead time, with the best possible quality and value... It provides guidance for the roles, responsibilities, artifacts, and activities necessary to achieve better business outcomes” ([14], p. 1). SAFe combines Agile, Lean product development and systems thinking. It is used to synchronize Agile teams. It distinguishes various configurations ranging from Essential SAFe containing the basic building blocks and most critical elements, to Full SAFe that is used to build large solutions and contains the levels of team, program, large solution and portfolio. One of the framework's fundamental principles is to decentralize decision-making. Arguments for this are among others that decentralized decision-making reduces delays and improves quality of decision where local knowledge is involved. This is not to say that central decisions are not necessary at all. However, they are far less frequent and limited to strategic decisions, i.e. decisions that have far-reaching impact and are outside the scope or knowledge of the teams. Examples of such decisions are decisions about product strategy or the standards to be used.

SAFe aims to break the functional silos by creating teams of Agile teams in the form of an Agile Release Train (ART). An ART is a networked organizational structure that is self-organizing and self-managing. In the ART teams, key stakeholders and other resources work toward an important ongoing solution mission. They share a single vision, roadmap and program backlog. ARTs deliver features (user functionality) and enablers (technical infrastructure).

The ART is positioned at program level. At this level, the role of the system architect is recognized. The system architect defines the overall architecture for a system, helps define nonfunctional requirements, determines the major elements and subsystems, and identifies the interfaces and collaborations among them. Above the program level, SAFe distinguishes two other potential levels: large solution and portfolio. The large solution configuration is meant for complex solutions spanning more than one ART. At this level, the solution architect is the one that defines a common technical and architectural vision for the larger solution. The portfolio configuration aligns portfolio execution, organizing development around one or more value streams. At this level, the enterprise architect works across value streams and programs to provide strategic technical direction and often acts as the epic owner for enabler epics.

In SAFe, architecture is not regarded as something that must be defined centrally in a top-down fashion. Instead, there is a need to balance emergent design and intentional architecture. Emergent design allows designers to respond to immediate user needs, intentional architecture takes care of matters that are beyond the scope of the team and provides guidance for inter-team design and implementation synchronization. Intentional architecture constrains emergent design, but only to the extent necessary, leaving enough room for maneuver for the teams. Emergent design corrects and feeds intentional architecture. Emergent design and intentional architecture together build the Architectural Runway, the technical foundation for future creation of business value. This only works if there is true collaboration between teams, architects and product management.

SAFe has a very different approach to enterprise architecture than TOGAF, giving it a more supportive role instead of a controlling role as is the case in TOGAF.

### 3.4 Thinking in services

An increasingly important concept within architecture is the concept of service. A service is a discrete unit of functionality that is well-defined, self-contained and remotely accessible, and does not depend on the context or state of other services. Services are defined by their interface, abstracting away the underlying implementation, which makes it easier to change the implementation of a service without disabling its use. The collaboration of services is often done by an orchestration function. Services can be internal to the enterprise or external. External services may reside in the Cloud. A service registry enables the publication and retrieval of available services. The concept of service can be applied at various levels, ranging from business services such as on-line payment services to application services such as financial administration to infrastructure services such as storage space. A Service-Oriented Architecture (SOA) consists of services that jointly achieve specific results [15].

A recent further development on the service-oriented architecture is the microservices architecture [16]. A microservices architecture structures an application as a collection of loosely coupled microservices. Microservices are services organized around business capabilities that can be independently developed and are independently deployable and scalable. Scalability is achieved by running multiple instances of a microservice. There are various infrastructural solutions for this, such as containers or serverless deployment. Microservices can be implemented using different programming languages, databases, hardware and software environment per microservice, depending on what best fits the specific microservice.

SOA, and its successor microservices architecture, has gained great popularity over the years because of its modular approach which provides the potential for flexibility. Therefore, it is very suitable to enable a sourcing strategy.



### 3.5 Enterprise architecture and digital sourcing

Digital sourcing is about the ability to flexibly combine digital capabilities provided by different parties, in offering propositions to the market. These digital capabilities, offered as services, can be of very different types, varying from managed applications in the Cloud (SaaS), providing scalability, to the delivery of back-office business services, such as salary administration, providing efficiency, to the provisioning of new customer interaction, by data-analytics, gamification and/or chatbots, providing better service to customers, and everything in-between. Enterprises must decide on their position concerning digital sourcing: where do they stand on outsourcing, co-sourcing and insourcing? This is a strategic decision. It relates to the prime value proposition of the enterprise: what unique value does the enterprise offer to the market? And what is needed to deliver this value in terms of competences and capabilities? A complicating factor is that it is increasingly hard for enterprises to define this value on their own. Instead, consumers determine value to an increasing extent [17]. The better the enterprise architecture enables flexible coupling and decoupling of capabilities, the better the enterprise can switch between capabilities and make just-enough, just-in-time use of capabilities offered in the market.

Flexibility in sourcing is an extremely important aspect of modern enterprises, as it determines how fast an enterprise can implement new business models and new manners of collaboration with others. Over the years, the role and conduct of enterprises in the market has changed. Several large companies that traditionally dominated a substantial part of the market got into trouble. Nokia, for instance, dominated a large part of the mobile phone market, but did not realize fast enough that their customers started to rate design above functionality and price. And it lost market share to Apple [17]. We also see that the average lifespan of companies is steadily decreasing [18]. Enterprises that try to do it all on their own are being replaced by enterprises that are open to cooperation with other enterprises [19]. Or enterprises that provide others the platform to cooperate, as is done by the new platform organizations.

The digital business strategies that are the result of rapid technological innovation lead to new value creation models. Pagani distinguishes three types of value networks [19].

The closely vertically integrated model is the classic value chain model. It is designed to centralize organizational intelligence. It is characterized by the presence of a (limited) number of giant components that are strongly connected in a sequential value chain. The driver behind this model is the need to achieve independence and achieve control over the entire value chain.

The loosely coupled coalition model emerged as a response to increasing market complexity because of incremental innovation. Instead of a singular value chain, a value network emerges with various kinds of partnerships between the different parties in the network. Usually some firms achieve more prominence and power by occupying a central position in the value network structure. They use their prominence to grasp a leadership role in pulling together resources. In other words, they take charge of network orchestration. This results in a value chain that is more disintegrated and open. The ability to connect effectively with others becomes a core competence.

The multisided platforms are the result of the emergence of cross-boundary industry disruptions. In contrast to the previous two models, the multisided platform is non-linear. In the multisided platform model, a company brings together two or more distinct groups of participants (sides) that need each other in some way. A multisided platform company builds an infrastructure (platform) that creates value by reducing distribution, transaction



and search costs generated when these groups interact with one another. Well-known examples of multisided platform companies are eBay, Visa and Booking.com to name but a few. Multisided platforms can have two, three, four or more sides. The more the sides, the higher the degree of complexity and the greater the challenge of balancing the interests of all sides. The choice of which side to sponsor or not is a fundamental strategic choice. Multisided platforms strive for network effects. Network effect means that the value of a service increases as more consumers use it or more suppliers augment it. In a world of digital and connected services, network effects are the key differentiator and driver of value creation. The multisided platform model may very well penetrate all industries in the near future. And with the multisided platform model new characteristics become important, such as openness, transparency and modularity.

Pagani argues that to survive in an increasingly uncertain and complex environment, organizations must transform their organizational intelligence into a new relational intelligence, enacting an open communication process with their stakeholders. For architecture, this implies a focus on enabling open communication and ensuring a modular structure.

Various factors play a role in catering for the flexible coupling and decoupling of resources or services. One determining factor is the granularity of the components on which the enterprise architecture is built. It is more difficult to outsource only part of a component than it is to outsource an entire component. For instance, if an enterprise has implemented an integrated ERP suite that supports sales, delivery, inventory management, payments and financial administration, it may be hard to outsource just the payments part.

Another factor is the ease with which a component can be extracted from its environment, i.e. how entangled it is with other components. Entanglement can occur horizontally and vertically. Horizontal entanglement means that components of similar nature are entangled. For instance, if the way a process is executed depends on the way another process is executed, or if the working of an application depends on the working of another application. Vertical entanglement means that components of different nature are entangled. For instance, the way a process is executed depends on the way a supporting application is designed. A component that communicates with other components by way of well-defined services can be outsourced easier.

Also relevant is the extent of standardization, especially the standardization of communication. Usually, an outsourced component does not function in isolation but exchanges information with other components. Broadly accepted communication standards enable such exchanges. A component that communicates by means of a widely recognized standard is easier to outsource than a component that communicates in an organization-specific manner. This holds not only for standardization of technical protocols, but, maybe even more importantly, also for semantic standardization. Within an ecosystem of collaborating parties, it is essential for smooth collaboration that participants understand each other. That is why many industry sectors develop common vocabularies, for instance in the water sector (Aquo), the building sector (building information modeling) and the care sector (HL7). Using such standards enables communication in precisely defined terms, making it easier to exchange information without misunderstandings or the need for complex translations, and thus to collaborate.

As technology becomes an important driver, the architecture must be able to absorb new relevant technologies [20]. Thus, the architecture must contain provisions to connect to new digital resources and services. Many organizations have been standardizing on technology for efficiency reasons. For core systems, this may still be a prudent way to go. For continuous delivery of digital services, however, organizations must be able to make use of the newest

digital resources, either as a service or as a technology. Adherence to strict standardization rules is not advisable. However, though standardization on technology may hamper innovation, standardization on interfaces stimulates it. Standardization on interfaces enables enterprises to use each other's services. It stimulates collaboration and interaction. It enables rapid innovation by combining existing building blocks in new ways. While standardization on technology is driven by efficiency, standardization on interfaces is driven by value delivery: it enables seamless interaction of capabilities from different sources to combinedly offer value that none of the collaborating parties could have offered on its own. Thus, a collaboration between Spotify and Uber enables Uber users waiting for their ride to personalize the music during their ride by selecting, via the Uber app, a Spotify playlist to be played during their ride.

Thus, we see that the enterprise architecture enables digital sourcing by (1) incorporating the right form of standardization, (2) defining the right granularity of components and (3) employing a form of service-oriented architecture. However, there is more to the picture. There are a couple of trends that further impact the enterprise architecture management function. These are discussed in the next three paragraphs.

### 3.5.1 Acting in an unpredictable context

Enterprise architecture looks at the bigger picture. It takes a holistic view of the enterprise—not only in terms of taking into account various facets of the enterprise, but also in terms of looking further ahead than the here and now. It is the aim of enterprise architecture to also cater for the future needs of the enterprise. This poses an increasingly hard challenge, as the future seems increasingly hard to predict. It is hard to capture beforehand all possible issues in a predefined set of detailed rules—especially where interaction with customers and partners is concerned.

Recently, awareness has grown that enterprises may need to differentiate between architecture regimes to cater for bimodal or multi-modal IT. Ross et al. argue that enterprises must distinguish between their operational backbone and their digital services backbone [21]. *The operational backbone* provides the resources for operational excellence. They define an operational backbone as the set of business and technology capabilities that ensure the efficiency, scalability, reliability, quality and predictability of core operations. Common elements of an operational backbone are master data management providing a single source of truth for key data that is to be shared throughout the enterprise (e.g. customer, order and product data), seamless and transparent transaction processing and standardized back-office shared services. This description of operational backbone is reminiscent of the concept of systems of record introduced by Gartner in their pace-layered applications model [22]. The important difference between the two concepts is the fact that Ross et al. are looking at it from a business perspective, i.e. sets of capabilities, whereas Gartner refers to applications. *The digital services backbone* facilitates rapid innovation and responsiveness to new market opportunities. Ross et al. define a digital services backbone as the set of business and technology capabilities that enable rapid development and implementation of digital innovations. Common characteristics of a digital services backbone are digital components including both technical services, like biometrics, and business services, like customer alerts, platform as a service—a technology hosting environment where the company can store and access large numbers of loosely connected microservices, repositories for collecting massive amounts of public (e.g. from social media), purchased, and/or sensor data, analytics engines for converting the above data into meaningful insights and connections to data and processes

residing in the operational backbone. This is reminiscent of the systems of innovation layer in Gartners pace-layered model—with again the distinction that the digital services backbone is viewed from a business perspective as a set of capabilities. Ross et al. argue that while the technological differences between the two backbones are likely to diminish with time, the need for their differing organizational characteristics will likely remain.

The implication for architecture is that it must enable the enterprise to quickly adjust to external developments and experimental results while at the same time maintaining robustness where necessary. One way to accommodate both these requirements is to vary the level at which architectural normative statements are made, i.e. the principle part of the enterprise architecture. In [23], it is argued that, just as is the case with regulations, architecture principles can be placed on a continuum from very abstract to very detailed (see for the distinction between rules and principles in the regulatory compliance context [24,25,26]). Principles are more abstract, expressing what is to be achieved and leaving room for interpretation. Rules are more concrete, expressing how things are to be done and leaving little room for interpretation. Rules are very useful in stable environments with a need for consistency and coherence—as may be the case in an operational backbone. Principles are more suitable in a dynamic environment, with a need for variation in responses, as in the digital services backbone. This is supported by the so-called knowledge-based view of the firm discussed by Grant [27]. Grant distinguishes four mechanisms for integrating knowledge: rules and directives, sequencing, routines, and group problem-solving and decision-making. Grant argues that rules and directives (i.e. written down directions) are suitable for communicating explicit knowledge among specialists and between specialists and non-specialists. Rules and directives are useful for tasks that are well-defined and to a great extent predictable. Group problem-solving and decision-making requires active interaction between participants and is needed for non-standardized tasks that are complex and unpredictable. Translating this to principles and rules, the more detailed knowledge expressed by rules is comparable to rules and directives and thus is useful for providing direction to relatively standard, predictable tasks. Principles, which are less precise, indicating desired outcome rather than the means to achieve that outcome, require more interaction and discussion in their application. They are the first choice when the organization is faced with disruptive technology.

Unpredictability is also apparent in the rapid decline in relevance of certain capabilities or the rapid emergence of unexpected parties exhibiting excellent capabilities. Examples are the decline of middlemen such as mortgage intermediaries and the rise of WhatsApp as an alternative to phone calls. Therefore, the enterprise architecture must be prepared for both decoupling and coupling of internal as well as external capabilities. This is done by carefully modeling the capabilities the enterprise needs, focusing on the correct scoping of capabilities and ensuring maximum independency between capabilities.

Unpredictability also necessitates a mindshift in the development of enterprise architecture. Refactoring should be a fundamental part of the architecture. This means, among other things, that architectural choices are evaluated for their retractability, that architectural choices are made at the latest possible moment instead of the first possible moment and that by rule architectural choices are provisional.

Summarizing, unpredictability requires fast reaction. However, not everything is equally unpredictable. Situationally applying a rule-based or principle-based approach provides the means to vary levels of freedom and hence flexibility. Careful scoping and designing of capabilities enables an enterprise to optimally exploit unpredictability in the relevance of internal capabilities and availability of external capabilities.

### 3.5.2 Acting in a value network

The boundaries of organizations are becoming less defined than they were and parties frequently switch roles. The same party may one moment act as client and the next as supplier—or competitor and client. For instance, an individual can both be a customer of bol.com buying books and a supplier offering their books through bol.com. Competitors and partners may emerge from unexpected directions. Only enterprises that are open to participation in fluid value networks will be successful. An example in case is the announcement of ING bank to become a platform organization and offer financial services from other financial institutions as well. This fluidity of roles has implications for the enterprise architecture.

A way of dealing with fluid value networks for enterprise architecture is to model the value network in terms of capabilities. From an enterprise perspective, the enterprise architect translates the ambitions and goals of the enterprise into the capabilities needed to realize these ambitions and goals. The next step is to model the way in which these capabilities collaborate. This collaboration can be modeled in terms of services delivered by one capability to another capability. Based on such a conceptual model, and dependent on the competences of the own organization as well as the interaction needs of using the services, decisions can be made on how to source each capability. If a party emerges that offers better services, i.e. excels in one of the capabilities, the organization may decide to switch.

A way to view collaboration between parties is the transaction-based approach on which DEMO (Design and Engineering Methodology for Organizations) is based [28]. DEMO breaks up a value chain or network into a tree of transactions between parties. A party in this case may be an individual, a business, a government organization or any other legal entity. A transaction consists of a request, promise, execution, delivery and acceptance. The transaction approach helps to clarify responsibilities. The promising party is the responsible party. Even if a party engages other parties to contribute to the execution of a transaction, it remains accountable to the asking party. Clear definition of transactions increases flexibility. In this sense, a transaction can be compared to a service. In both cases, a provider is responsible for the delivery of a well-defined unit of functionality to a consumer.

Summarizing, enterprise architecture can support participation in a value network by modeling it in terms of transactions between parties possessing valuable capabilities.

### 3.5.3 Acting in a customer-centric world

The rise of data-driven services offers consumers a lot of choice. Customers will no longer be loyal to enterprises that do not cater to their needs. In the past, as far as IT was concerned, enterprise architecture tended to focus on internal efficiency, whereas nowadays customer experience seems to be a main driver [30]. This asks for different architecture principles.

In the recent past, architecture principles about the use of IT were primarily focused on the values of robustness and efficiency [29,30]. This is illustrated by frequently used principles such as: (1) reuse before buy before build, (2) no redundancy, (3) shielding by firewalls and (4) standardization on technology. However, these principles are not aimed at providing an excellent customer experience. To do the latter, enterprises may need to experiment and take risks. They must learn to listen to their customers and start thinking outside-in.

Digital interaction processes offer lots of opportunities to get to know customers. However, enterprises must take care to use these data in a responsible manner and not to misuse the data they collect. This means being careful and transparent in how data are used and always keeping the interests and concerns of the customer in mind. In the architecture, this

emerges among others in the form of principles concerning the use of data. This applies not only to data about customers collected by the enterprise in the execution of its service delivery, but also to the use of widely available data. For instance, the use of machine learning in decision-making carries the risk of treating customers in a biased manner. All this may lead to new principles, such as: (1) development by value-sensitive design, (2) use of customer data based on informed consent.

Summarizing, enterprise architecture stimulates customer centricity by formulating architectural principles from an outside-in perspective in which internal efficiency is not the primary driver, but acting in the interest of the customer, providing true value to the customer.

### 3.6 Enterprise architecture management function

Enterprise architecture provides the organization with the overview and insights necessary to make informed decisions on how to structure its products and services, processes, information flows and technology in a way that is aligned with its ambitions and strategy. The enterprise architecture models provide the required overview and insights. The enterprise architecture principles provide guidance on what to do or not to do when designing changes. The enterprise architecture management function is concerned with effectively applying these models and principles. Effectively applying enterprise architecture models and principles means applying them to provide direction to the change processes of the organization. For it is in these change processes that design decisions are being made, either explicitly or implicitly. Decisions that we want to be informed by enterprise architecture.

Decisions take place at various levels within the organization, ranging from strategic decisions at board level to operational decisions at for instance software development level. Enterprise architecture is relevant to all these levels of decision-making, but in different ways. At board level, the role of the enterprise architecture is to provide overview and to inform management about the impact and consequences of certain choices. Also, we see enterprise architects increasingly take the responsibility to proactively inform management about opportunities, often generated by the architectural choices. At operational level, the role of enterprise architecture is mainly to provide guidance in making specific design choices, either by providing relevant rules or by providing advice on specific solutions [3].

The way the enterprise architecture management function is organized differs between enterprises. Depending on the size of the enterprise, the number of architects may range from one employee fulfilling the role of architect for a part of his/her time to dozens of architects of various types spread over different departments. A common distinction is between enterprise architects who maintain a holistic view over the entire enterprise and solution architects who apply this enterprise architecture to guide the design of specific solutions, for instance a new service to customers. Enterprise architects are usually centrally organized, either as a staff function below board level or as a central team within the IT department. Solution architects may be positioned within or near the development teams. The tasks of enterprise architects consist of keeping the enterprise architecture up to date and in line with the ambitions and strategy of the enterprise, developing roadmaps for realizing target architectures, advising the board about the applicability of technological innovations, implementing governance processes to ensure compliance with the enterprise architecture and educating the organization in enterprise architecture. Solution architects translate the models and principles of the enterprise architecture to specific guidelines for development teams, for instance in a so-called project-start architecture, model the architecture of specific

solutions in collaboration with designers, monitor whether design decisions are in line with the architecture and act as advisor to development teams in cross-team issues.

The enterprise architecture management function faces various challenges in the fulfillment of its role. First, there is the question of mandate. Usually, it is not the architect who makes the final decision. In this respect, it is essential that the architects are backed by management. All too often, architects feel forced to sell architecture to the organization as if they themselves are the primary beneficiaries of the enterprise architecture. The main beneficiary of the enterprise architecture should be the board, however. That is why it is important that the enterprise architecture management function reports to the board directly, preferably to the CEO. It is the CEO who should take ownership of the enterprise architecture and who must understand and subscribe all choices made therein. In practice, however, the enterprise architects often report to the IT manager or CIO, limiting their scope to IT. This may restrain their effectiveness in aligning the enterprise architecture with the business strategy.

Effective application of enterprise architecture is also strongly related to its embedding in strategic dialogue as well as in solution development. The enterprise architecture management function must ensure that they can provide relevant insights at the moment of decision-making. Sometimes that may be a vision paper on the applicability of Cloud computing for the organization, other times that may be a specific solution pattern for communication with partners. Sometimes the insights needed can be simply derived from the existing architecture models and principles and sometimes a dedicated deliverable will have to be made. Architectural deliverables must be tuned to the stakeholders of these deliverables and architectural processes must be integrated with the organizations change processes.

The discipline of enterprise architecture is still relatively young. Different organizations show different levels of maturity of the enterprise architecture management function. As a discipline, it is far less established as for instance project management. The demands of digital business strategy and dealing with unpredictability, however, require a high architecture maturity level—especially, because the demands of fast response imply that architects cannot always fallback on premeditated models and visions, but need to “think on their feet” more.

Enterprise architecture maturity models are a means to support organizations in developing their enterprise architecture management function. Maturity models are conceptual models based on the idea that organizational competences develop through a number of anticipated, desired or logical stages from an initial state to a more mature state [31]. More mature means better equipped to fulfill its purpose. The basic components of a maturity model are (i) a number of overall maturity levels, (ii) a number of aspects or areas that can be developed along a predefined evolutionary path to achieve the defined maturity levels and (iii) descriptions of each step on the evolutionary path. In addition, a maturity model may contain suggestions on how to perform the various steps in terms of improvement actions.

Maturity models can be used with three different objectives in mind. First, they can be used to assess the current state of the architecture practices, i.e. how well is the enterprise architecture management function organized and how efficient or effective is it? What are its weak and strong points? Second, maturity models can be used for benchmarking, i.e. to make comparisons between enterprises. Of course, this objective can only be realized if all enterprises use the same maturity model and are willing to share their maturity scores. Third, maturity models can be used to improve the enterprise architecture management function. Dependent on an assessment of the current state, a maturity model may suggest adequate measures to grow to the next maturity level.



An example of a maturity model is the model developed by Ross et al. [32]. They discuss four stages that an organization passes through, making increasingly effective use of enterprise architecture: (1) business silo (IT applied to specific business needs, locally optimal business solutions), (2) standardized technology (central technology infrastructure), (3) rationalized processes (base of IT-enabled processes for core operations, shared and standardized business processes and/or data) and (4) business modularity (building on the core processes with plug-and-play processes built internally or externally). As an organization moves up through the stages, the strategic impact of IT increases. The organizational changes at each stage include new business processes, new management practices, new governance approaches and new attitudes about the role of IT. Ross et al. found in their research that large firms required on average five years per stage. They issue a warning that it is not possible to skip a stage. Each stage involves technology and organizational changes that prepare for the next stage. As the term suggests, to fully support a digital business strategy seems to require the stage of business modularity. Some organizations, however, are still struggling to get their operational backbone up and running, i.e. to reach the stage of rationalized processes. This is a serious threat to their business success. The maturity model of Ross et al. is primarily useful for assessing the current state of the architecture practices.

Another type of maturity model is the Dynamic Architecture Maturity Model, DyAMM, which is part of DYA [33]. DyAMM is designed and used for all three objectives mentioned above. It could be used to grow to level four of Ross et al. The DyAMM model distinguishes 17 aspects, called focus areas that together comprise an enterprise architecture practice. The focus areas are the development of architecture, use of architecture, alignment with business strategy, alignment with realization, relationship to the As-Is state, responsibilities and authorities, alignment with change portfolio, monitoring, quality assurance, management of the architectural process, management of the architectural products, commitment and motivation, implementation of the architectural role, architectural method, interaction and collaboration, architectural tools, budgeting and planning. The aim of DyAMM is to assist enterprises in increasing their enterprise architecture maturity step by step by developing each of the focus areas in a balanced manner. The idea behind DyAMM is that though each of the 17 focus areas must receive attention, this does not mean that each must be given equal consideration at the same time. For instance, not every focus area is equally relevant at the start: the use of architectural tools will certainly become a key concern at some point, but enterprises that are still in the phase of building up an architectural practice can focus more productively on aligning the architecture with the business strategy. Tools will have their turn. Furthermore, any given focus area need not be brought up to its full state of development right away. Each focus area has its own levels of maturity and the DyAMM model provides a matrix that shows how to alternately develop the different focus areas level by level. DyAMM distinguishes 12 overall maturity scales. For most enterprises, however, scale 6 will be more than adequate. Enterprises with a complex product portfolio and corresponding complex organization may aspire to scale 8 or even 10.

There is not one best way to organize the enterprise architecture management function. Its effectiveness very much depends on whether its implementation matches the size and organization of the enterprise. However, a maturity model such as the DyAMM model provides a useful tool to assess and subsequently develop the enterprise architecture management function to the required level.



### 3.7 Conclusion

The aim of enterprise architecture is to provide the structure to an enterprise that matches its strategic ambitions and enables the enterprise to turn strategy into execution. It does so from a holistic perspective, aiming for the seamless connection of products and services, processes, information flows and technology. The discipline of enterprise architecture consists of both the enterprise architecture artifacts, distinguishable into models and principles, and the enterprise architecture management function of developing and applying the enterprise architecture artifacts.

Digital sourcing in today's world puts certain requirements on enterprise architecture. Above all, it asks for the flexibility to quickly respond to changes in the environment and to flexibly start and end collaborations with other parties.

The enterprise architecture is an important enabler of digital sourcing by applying a service-oriented approach, modeling independent business capabilities at the right level of granularity, standardizing on interfacing instead of technology and formulating architectural principles from an outside-in, customer-centric perspective. The balance between robustness and flexibility can be supported by varying the level of normative statements using principles on one occasion and rules on another.

To be effective, the enterprise architecture management function must be aligned with the change processes in the organization, at all levels from strategic to operational, to be able to provide to-the-point advice at the moment of decision-making. This implies that the enterprise architecture management function tunes itself to the change processes in which the design decisions are made. In this effort, maturity models may function as useful checklists.

### References

1. The Open Group: TOGAF version 9, p. 5 (2009).
2. McDonald, M.P.: The Enterprise Capability Organization: A Future for IT. *MIS Quarterly Executive*, 6(3), 179–192 (2007).
3. Berg, M. van den, Steenbergem, M. van: *Building an Enterprise Architecture Practice*. Springer, Dordrecht (2006).
4. Ward, J., Peppard, J.: *Strategic Planning for Information Systems*. 3rd edition, Wiley & Sons, Chichester (2002).
5. Versteeg, G., Bouwman, H.: Business Architecture: A New Paradigm to Relate Business Strategy to ICT. *Information Systems Frontier*, 8, 91–102 (2006).
6. Schekkerman, J.: *How to Survive in the Jungle of Enterprise Architecture Frameworks.*, 3rd edition, Trafford Publishing, Victoria, BC (2006).
7. Zachman, J.A.: A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3), 276–292 (1987).
8. The Open Group, <http://www.opengroup.org/TOGAF-9.2-Overview>
9. Lankhorst, M., et al.: *Enterprise Architecture at Work*. Springer, Heidelberg (2005).
10. Wagter, R., Berg, M. van den, Luijpers, L., Steenbergem, M. van: *Dynamic Enterprise Architecture: How to Make It Work*. Wiley, Hoboken (2005).
11. Wagter, R.: *Enterprise Coherence Governance*. Doctoral Thesis. Scholars' Press, Cambridge (2015).
12. Poort, E., van Vliet, H.: Architecting as a Risk- and Cost Management Discipline. Ninth Working IEEE/IFIP Conference on Software Architecture, 2–11 (2011).
13. SAFe. <https://www.scaledagileframework.com/agile-architecture/>.
14. SAFe 4.5 Introduction – Overview of the Scaled Agile Framework for Lean Enterprises. A Scaled Agile, Inc. White Paper (2017).
15. Berg, M. van den, Bieberstein, N., Ommeren, E. van.: *SOA for Profit, A Manager's Guide to Success with Service Oriented Architecture*. IBM, Sogeti, Paris (2007).
16. Amundsen, M., McLarty, M., Mitra, R., Nadareishvili, I.: *Microservice Architecture – Aligning Principles, Practices, and Culture*. O'Reilly Media, Portland, OR (2016).

17. Keen, P., Williams, R.: Value Architectures for Digital Business: Beyond the Business Model. *MIS Quarterly*, 37(2), 642–647 (2013).
18. Anthony, S.D., Viguierie, S.P., Waldeck, A.: *Corporate Longevity: Turbulence Ahead for Large Organizations*. Innosight, Executive Briefing, Spring (2016).
19. Pagani, M.: Digital Business Strategy and Value Creation: Framing the Dynamic Cycle of Control Points. *MIS Quarterly*, 37(2), 617–632 (2013).
20. Bharadwaj, A., El Sawy, O. A., Pavlou, P. A., Venkatraman, N.: Digital Business Strategy: Toward a Next Generation of Insights. *MIS Quarterly* 37(2), 471–482 (2013).
21. Ross, J. W., Sebastian, I. M., Beath, C., Mocker, M., Fondstad, N. O., Moloney, K. G.: Designing and Executing Digital Strategies. Proceedings of the International Conference on Information Systems (ICIS), 1–16 (2016).
22. Mesaglio, M., Hotle, M.: Pace-Layered Application Strategy and IT Organizational Design: How to Structure the Application Team for Success. Gartner, Stamford, CT, October 5 (2012).
23. Eusterbrock, T., van Steenberg, M.: Principle-Based Approach in Enterprise Architecture practice; Finding the Sweet Spot, (2016). [http://www.dya.info/sites/dya.info/files/attachments/1602028%20DYA%20Principle-based%20approach%20Enterprise%20Architecture\\_07.03.2016\\_0.pdf](http://www.dya.info/sites/dya.info/files/attachments/1602028%20DYA%20Principle-based%20approach%20Enterprise%20Architecture_07.03.2016_0.pdf).
24. Black, J., Hopper, M., Band, C.: Making a Success of Principle-Based Regulation. *Law and Financial Markets Review* 1, 191–206 (2007).
25. Black, J.: Forms and Paradoxes of principles-Based Regulation. *Capital Markets Law Journal* 3(4), 425–457 (2008).
26. Burgemeestre, B., Hulstijn, J., Tan, Y. Rule-Based Versus Principle-Based Regulatory Compliance. Proceedings of the 2009 Conference on Legal Knowledge and Information Systems, JURIX 2009: The Twenty-Second Annual Conference, 37–46 (2009).
27. Grant, R.M. Toward a Knowledge-Based Theory of the Firm. *Strategic Management Journal*, 17, 109–122 (1996).
28. DEMO, <http://www.ee-institute.org/en/demo>
29. Boucharas, V., Steenberg, M. van, Jansen, S. Brinkkemper, S.: The Contribution of Enterprise Architecture to the Achievement of Organizational Goals: A Review of the Evidence. In: Proper, E., Lankhorst, M.M., Schönherr, M., Barjis, J. and Overbeek, S. (Eds.), *Proceedings of the 5th International Workshop on Trends in Enterprise Architecture Research*. Springer, Berlin Heidelberg, 1–15 (2010).
30. Plessius, H., Steenberg, M. van, Slot, R.: *Perceived Benefits from Enterprise Architecture*. Eighth Mediterranean Conference on Information Systems, Verona (2014).
31. Gottschalk, P. and Solli-Saether, H.: Maturity Model for IT Outsourcing Relationships. *Industrial Management & Data Systems*, 106(2), 200–212 (2006).
32. Ross, J.W., Weill, P., and Robertson, D.C.: *Enterprise Architecture as Strategy*. Harvard Business School Press, Boston, MA (2006).
33. Steenberg, M. van, Bos, R., Brinkkemper, S., Weerd, I. van de, Bekkers, W.: Improving IS Functions Step by Step: The Use of Focus Area Maturity Models. *Scandinavian Journal of Information Systems*, 25(2), 35–56 (2013).