Chapter 5
The Process of Enterprise Architecting

5.1 Introduction

As we have seen in Chapter 4, the purpose for which a particular architecture is created largely determines the kinds of results needed. This purpose also impacts on the form and execution of a good architecture *process*. However, the purpose is not the only major influencer. Another major influencer is the *multitude* of stakeholders, which over time need to agree on the direction of the enterprise transformation. In this change process the insights gradually evolve, while the decisions taken during the process may change the direction of this process. Therefore, the architecture process can also not be a linear one. It strongly depends on situational factors, and therefore cannot be served by a one-size-fits-all approach.

The core process of enterprise architecture encompasses creating, applying and maintaining the architecture for its intended purposes. In the way this process is executed, we see best practice patterns and styles emerge, sometimes even materialising in terms of "architecture schools". Whether a specific style or pattern is effective or not, not only depends on the purpose of the architecture, but also on the architecture maturity of an organisation, it's management style and culture. Process quality criteria, mainly derived from purpose and maturity, are required to enable a choice for an effective and efficient architecture process. The maturity, the situational criteria and preferred architecture patterns will all influence how to organise the continuous improvement of the architecture function.

The remainder of this Chapter is structured as follows. We start by describing the core elements – *create, apply and maintain* – of the process of enterprise architecting. Next, we will give examples of patterns, as well as best practices, to execute this core process. Subsequently, we will reflect on the importance of architecture maturity, introducing an architecture maturity model. We then continue by showing how the core architecture process can be organised to implement the *plan – act – learn* cycle of the enterprise as a whole, taking outsourcing of architecture roles as an option. Finally, we summarise the architecture process concepts introduced and highlight their coherence.

5.2 The core process of enterprise architecting

Enterprise architecting involves a number of core processes. First of all an enterprise architecture has to be *created*. To be really useful as an instrument within an enterprise, the architecture has to be *applied* in line with its purpose. To remain useful in a changing world, an enterprise architecture should be *maintained* as well. Each part of the architecture process – *create, apply and maintain* – will to a large extent be influenced by the purpose of a specific architecture.

5.2.1 Creating enterprise architecture

Let us first elaborate on the act of creating an enterprise architecture. As we have seen in Chapter 4, the purpose of enterprise architecture will determine the results which should be produced. In general, both tangible and intangible results will be strived for, each requiring its own process. What does such a process look like?

Consider the following sketch of an enterprise architecting process, mimicking a basic project setup:

> *Inspired by Chapter 4, one would start by selecting the (tangible and intangible) results needed. Then one would carry on by defining the activities needed to produce these results and order them in a plan. Subsequently, one would have to arrange staffing of the project and request formal permission to initiate the project. Finally, one would disappear out of sight, and re-appear about a half year later with beautiful principles, models and views.*

This is a rather naïve approach, which is deemed to be totally ineffective. To indeed realise the intention of enterprise architecture – enabling the steering of change – much more is needed. For instance, a shared conceptualisation should emerge among stakeholders about:

- a **to-be** situation,
- the **as-is** situation,
- any constraints that should be met,
- purposes of the enterprise architecture that are met and those that are as yet unmet.

Even more, the concept of enterprise architecture might be a new means to the organisation, in which case an introduction is called for. In the organisation several large cultural differences may exist as well, e.g. between *thinkers* and *do-ers*, between *domain experts* and *management*, between *sales* and *operations*, et cetera. This all opens up new requirements about how to communicate about, and arrive at, enterprise architecture results, about the benefits of (an) enterprise architecture and about the process which is followed in creating enterprise architecture. For this kind of change you have to involve, get access to and get buy-in from important stakeholders before, during and after the "enterprise architecture project". Communication is an important aspect in this.

This raises the question: *Which stakeholders are required, how and when should they be involved, and how (and to what extent) does this depend on the purpose of the architecture and context in which it will be created*? Best practices learn that it is important to deal with the multiplicity of stakeholders and their concerns at the right time. Be aware that stakeholder needs and expectations may change over time. Even more, the set of stakeholders involved may change over time as well, more or less necessitating a (time-boxed) iterative approach. We limit ourselves to a few examples:

- Suppose an enterprise architecture is mainly used for decision making on an intended business transformation. Then a **to-be** *description*, possibly in the form of an sketch, is probably more relevant than specific guidelines on how to arrive at the **to-be** situation. This in turn means that mainly managerial and context stakeholders need to be involved in a more brainstorming way of working, in order to achieve common understanding and commitment.
- Assume an enterprise architecture has to be created in a context where "*don't ask why but ask how high*" is the predominant attitude. Instead of consultative and consensus building workshops, the way of working here will include more directive and elaborating workshops.
- Envisage an organisation for which enterprise architecture is a new endeavour. Then the enterprise architecture team should consider to keep the more complex and abstract intermediate enterprise architecture results for internal use only and to communicate with all stakeholders in terms of the added value of applying enterprise architecture.
- The concern of the IT manager about which IT to procure, is valid and should be dealt with separate (generally: later) from formulating business strategy impact.
- The concerns of the security officer regarding the potential impact on privacy regulations, when storing customer data, should be included right from the beginning.

In shaping the creation process, of course more general project and change management criteria apply as well [52]. As discussed in Chapter 4, the purpose of the enterprise architecture determines the level of quality required from the deliverables.. Together with constraints in duration, net time or money, this in turn will determine the feasible level of detail and accuracy. In deciding which staff members should be involved, we should now explicitly include criteria not only about knowledge but also on position, influence and stakeholder network.

5.2.2 Applying / using enterprise architecture

We now turn our attention to the application and actual use of enterprise architecture. In Chapter 3 we discerned a regulation-, a design- and a pattern-oriented perspective on enterprise architecture. In Chapter 4 we elaborated this into four main goals of enterprise architecture: *deciding*, *specifying*, *informing* and *contracting*. In each

of those four main goals, a shared conceptualisation is needed on issues such as opportunities and problems, strategies, possible solution alternatives and their trade-offs, and finally the chosen solution alternative. Each of those goals of enterprise architecture has their specific needs for a process.

We start by looking at situations where enterprise architecture is used to support decision making. To be effective as a steering instrument, enterprise architecture should be embedded in the "ordinary" steering processes of an organisation, for which Figure 5.1 shows an example. In the phases of *drafting needs* and *business case*, we use architecture mainly for the purpose of shared conceptualisation in terms of principles and a high level design. This helps in scoping the ambition, overseeing complexity and risks, and finally deciding if the organisation actually wants to execute the depicted change, i.e. answering the question: "*do we want to, and are we able to, do this?*" Typically, the latter decision will be taken by senior management and key stakeholders. When a positive decision has been made, the next question is how to realise and control the intended change. In this phase, enterprise architecture helps to plan the change and ensure compliance with the principles. In trading off solution alternatives, it will appear that not all principles can be fully met in any of them. Take for example the principles "service oriented" and "proven solutions", then in selecting an alternative with a higher priority for proven solution, the enterprise architecture will help to (1) make explicit the consequence of "less service oriented" in the solution and (2) give additional guidelines necessary for contracting, e.g., prescribing service layering or legacy wrapping. All along the way, enterprise architecture will thus enable risk assessment and mitigation.

Fig. 5.1 Enterprise Architecture embedded in the steering process of an organisation

A specific use of enterprise architecture in decision making we find in the planning of change at several levels in the enterprise. Often enterprise architects collaborate with program managers in the long-range planning of transformations, e.g. in the context of a *design authority*. This results in a *plateau planning*, defining plateaus as a steady state of the business, and projects to arrive at these plateaus. Enterprise architecture will help making the plateau descriptions complete as far as content is concerned, at the same time defining project scopes and project dependencies, thus enabling estimations of time, money, risks and feasibility. In the "Perla del Nord" case as introduced in the previous Chapter, we actually focused on the *create* process and did not really cover the *apply* and *maintain* processes. In

this Section we will remedy this, also exemplifying these processes. In the "Perla del Nord" case, a possible plateau could be a situation in which the pizzeria is able to act in the B2B-market, including the requirements on B2B-contracting and credit risk checking. Another plateau could be the broadening of the product portfolio with e.g. lasagne. Senior management can now easier decide on ambition levels of plateaus, while program management can decide on how to arrange their program in terms of well scoped and feasible projects. In the example of "Perla del Nord", such projects could be "*marketing B2B service offering*", "*make available baking car/bus*", "*enable automated credit checking*", "*authorisation of B2B-employees*" and "*multi-channel offering for B2B*". The combined insights per intended program in turn can support the enterprise portfolio management in its investments in enterprise assets, aligning the initiatives, preventing overlaps and choosing the order of programmes. Summarising, the process of decision-making requires from enterprise architecture timely and stepwise growing insight in consequences of decisions. This insight to enables *informed* trade-offs in terms of time, money, risks and feasibility, i.e. laying the foundation for *informed governance*. This process of decision-making is quite diverse: in several phases, different stakeholders need to be involved, also depending on the outcome of trade-offs made in between.

Special attention is needed to make the link between enterprise architecture and programmatic steering of change (see Subsection 2.4.2) effective. This should already start at the level of *governance*. Enterprise architecture supports corporate governance by giving insight into substance and coherence of the entire value chain. At the same time, architecture itself needs to be embedded in the overall change and governance processes of the enterprise. *Projects* aim to realise parts of the to-be situation, in which they have to comply with the guidelines and the structure imposed by the enterprise architecture. To aid projects with this compliance, they should be accommodated with selections from the enterprises architecture, which are relevant to the scope of the project, additional viewpoints suitable to the concerns of the project's stakeholders, as well as operational criteria to ascertain the compliance of the project's result to the enterprise architecture. TOGAF's phase G [139] specifies this connection between the enterprise architecture and a specific project in a so-called *architecture contract*. Subsequently, the project has the responsibility (and freedom) to develop the architecture for their solution within the context of the architecture contract. Other sources use terms such as Project Start Architecture (PSA) to refer to the same connection [41, 148]. *Portfolio management* can use enterprise architecture as a common language to coherently define the programmes needed. Indeed, the enterprise architecture will show which intended components contribute in what extent to which goals and strategy, enabling underpinned choices in adding or removing parts of the organisation and technology. At the same time, enterprise architecture needs an overview of the programmes in the portfolio as an input. Finally, where *programme management* focuses on the managerial aspects of a body of projects, enterprise architecture ensures the cohesion between the product aspects of these projects.

We now turn our attention to the role which enterprise architecture can play in restricting and guiding design freedom, as required in specifying and contracting.

This role requires processes to formulate these restrictions and to ensure the compliance to them. In formulating the design restrictions, views are made for a part of a transformation, mainly indicating relevant principles and scoping in terms of sub-systems (component-systems and/or aspect-systems).

Suppose, "Perla del Nord" has decided to strive for the first plateau, i.e., to indeed enter the B2B-market. For the project "*make available baking car/bus*", such design restrictions could take the form:

- relevant principles:

  - bake while driving,
  - ability to handle growth in volume of a single delivery is more important than cost-efficiency,

- scoping of systems:

  - subsystem: the (combined) actor role of baker and deliverer, which we referred to as (functionary type) *transporter* in Chapter 4, including his cooperation with cook and order taker,
  - included aspects systems: vehicle (physical transporting & baking device) to support the transporter, including the information supply about the vehicle (location tracking, maintenance-guidelines for the driver),
  - excluded aspect systems: the recruitment / education / job descriptions for the transporter.

For the project "*multi-channel offering for B2B*", such restrictions could be:

- relevant principles:

  - all usual communication technologies for our customers should be supported,
  - re-use previously made models,

- scoping of systems:

  - subsystem: the actor role of the completer, as implemented in the functionary type *order taker*, including his cooperation with customer, cook and transporter,
  - included aspect systems: internet-channel & voice-channel, including educating the order taker in the use of that,
  - excluded aspect systems: recruitment / job descriptions for the completer.

Note that designers have freedom within the indicated restrictions, e.g., in the technology to be used for the voice channel. In communication with the marketing department of "Perla del Nord", it could well turn out that the *"usual communication technology"* for the target group is only instant messaging (IM) or Skype. This could mean that in "*implementing the internet channel*", e-mail will not be included.

In this last example, we see how a typical design decision emerges, which requires an iterative designer between designer and enterprise architect. Such a communication will result in four possible conclusions:

1. the enterprise architect changes (the project view on) the enterprise architecture, e.g. by including extra (new or already existing) principles or models, or by formulating better constraints;
2. the designer changes the intended design decision to adhere to the given design restrictions;
3. the enterprise architect decides to allow the decision made by the designer, at the same time adding this precedent to the enterprise architecture (as a kind of jurisprudence);
4. enterprise architect and designer agree that an intended design decision indeed has to be taken on project level, so that project and architecture align or at least do not contradict.

In the example of "Perla del Nord" this could mean that:

1. the enterprise architect informs the designer about already chosen but by accident not in the project-view included IM-standards, VoIP-standards and the data model for (retail) *client*,
2. the enterprise architect specifically emphasises the principle "we choose for open standards" and therefore the designer concludes that he will seek for a Skype-alternative,
3. since at this time no open standard based B2B-used alternative for Skype is available, the enterprise architect accepts and documents the proposed design decision to use Skype, and
4. the enterprise architect agrees that the intended decision not to implement customer-facing e-mail is compliant with the enterprise architecture indeed and can be taken at project level.

Since a guaranteed complete view for a project does not exist, this iteration helps the enterprise architect in finding a balance between over- and under-specifying. In order to ensure compliance, a compliance measurement process such as a formal audit or periodic review needs to be in place.

The created enterprise architecture is of interest to a wide variety of stakeholders. At the same time applying architecture is part of a change process, since this new means will impact the way of working for many. Therefore, on top of the communication before and during creating enterprise architecture (see Subsection 5.2.1), further communication is needed. To create additional buy-in, the communication should start at the level of awareness, to let stakeholders know that an enterprise architecture is in place and what benefits they generally can expect from it. Once awareness is in place, further communication can be targeted more specifically to the different stakeholders. For instance, decision makers need to understand the benefits, the typical insights they can expect from using it and the type of decisions it will facilitate. Project leads and lead designers need to understand the impact and value of expected restrictions at project level for the enterprise as a whole and the interaction with enterprise architects they can expect. As another example, the enterprise architecture can demonstrate its added value as a shared conceptualisation of the **to-be** situation and the transformation required. In the end, informing about enterprise architecture is analogous to normal communication: decide on target group and the

message per target group and subsequently derive the means to be employed (video, intranet, posters, road show, conference). Apart from this one-off communication, enterprise architecture also ensures the required continuous flow of information by giving and stopping access to enterprise architecture materials.

As we have seen in this Section, the different ways of applying enterprise architecture give a different emphasis in processes. For decision making the emphasis is on supporting, for ensuring compliance – e.g. in specifying and contracting – enterprise architecture will direct and in informing communication is paramount.

5.2.3 Maintaining enterprise architecture results

We now finally turn our attention to the act of maintaining enterprise architecture results. By maintaining the enterprise architecture we will understand both *monitoring* (business and/or technology) changes that might be relevant and *updating* the enterprise architecture. The effect of maintaining is that architecture results continuously and adequately reflect "reality" to a known and controlled extent. Now why and when is such maintenance of an enterprise architecture necessary? And how should it be done? Instead of giving a detailed description of the general ways for change management and impact analysis, we give some guidelines to adapt some common procedures. Sometimes, the creation of an enterprise architecture is a one-off event, e.g., when deciding not to go ahead with an intended change. The only relevant "maintaining" here would be to make this a conscious decision and enable traceability of this decision. Most of the time, however, such "creating of an enterprise architecture" is not a one-off event but rather an integral part of an overall (continuous) change process of an enterprise. Enterprise architecture results, in themselves, have a stable character and – because of the investments made in terms of involvement of stakeholders, time and money – represent a serious asset of the enterprise, deserving to be leveraged. A monitoring activity should therefore (regularly) assess whether the enterprise architecture should be updated by estimating the impact of typical change drivers. For instance, to what extent is the business changing, internally or externally: when should new legislation or supervisory guidelines be applied, when will innovations become available, are new goals pursued for, is a new strategy under development, are new products and services on their way? Or take the availability of people and means: will certain capabilities of people be less available, will capabilities become cheaper available elsewhere, is new technology or the phasing out of existing technology announced? Also internal reasons can be a change driver, such as the desire for a different and more efficient process, signals of issues/defects reported from projects or discussions with these projects. In the case of pizzeria "Perla del Nord", the project has chosen IM and Skype as technology for the voice channel, which raises several questions for EA such as "also use these channels for internal communication"; this means additional guidance is needed at the enterprise level, e.g. to prevent overlapping technology choices for IM. And of

course the stakeholders can change, by person or by role, which may be reason to see new concerns introduced and to give other concerns a different priority.

The monitoring task thus reveals the need to adapt the enterprise architecture. As a next step, the size and nature of the required adaptation has to be assessed: is this a minor change, to be solved by a well-localised update, or is this a major change leading to a new version, to be realised by a partial or complete (re-)creation of the enterprise architecture? A best practice rule of thumb is the amount of stakeholders involved: with one stakeholder it probably can be solved by an update. In the case of such an update, models and viewpoints will be improved and accentuated, as well as adding decisions to waive the enterprise architecture as a precedent to the documentation. Finally, the changes will be communicated to the relevant stakeholders.

When a major change is required, one should approach this as a (re)creation. In other words, all things stated before concerned the creation of an enterprise architecture applies: change management should start, the right stakeholders should be involved, etc.; a new architecture evolution cycle is started.

5.3 Patterns for enterprise architecting

For the core architecture processes we just introduced, several approaches with a proven track record have emerged. We will discuss their characteristics, reflecting on the assumptions under which such approaches can be successful. This will underpin our proposition that there is no one-size-fits-all approach to architecting, and that a situational approach is needed.

5.3.1 Architecture process patterns and architecture schools

We start with **TOGAF** [139], which contains the Architecture Development Method (ADM). ADM (see Figure 5.2) "*describes a method for developing an enterprise architecture*", in our terms for the architecture core process, including maintenance (called by TOGAF/ADM: phase H - Architecture Change Management). For each phase, ADM provides several guidelines and best practices. TOGAF/ADM does not prescribe any set of specific enterprise architecture deliverables; therefore it may be used in conjunction with the set of deliverables of another architecture framework. ADM describes a phasing model, starting from Framework and Principles via Architecture Vision (phase A) and Business Architecture (phase B) all the way to Migration Planning (F) and Implementation Governance (G). ADM points out that it can be applied iteratively in the whole process, between phases, and within a phase. ADM is continuously driven by the requirement process (the circle at the heart of Figure 5.2).

The Business-oriented Method for Information-planning (BMI) as described by [119] emphasises quality of models as a basis for long-term and high-quality
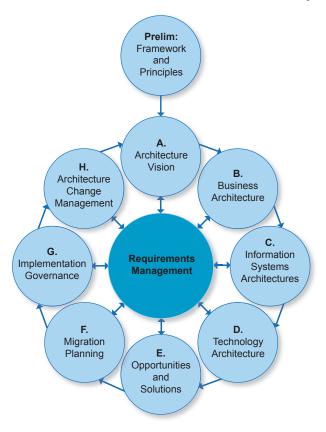
Fig. 5.2 Architecture Development Cycle

structure of the information supply. BMI has two central paradigms, namely (1) the distinction between essence and implementation of an organisation and (2) the distinction between business models and information models. This also determines the way of working: creating architecture begins with business models – preferably modelling the essence of the organisation first – and ends with the two types of information models and its interconnections. The way of working is focused on creating architecture and leaves much freedom for the way of interacting with the stakeholders.

The Design & Engineering Methodology for Organisations (DEMO) as described by [35] also emphasises quality of models to underpin high-quality decisions to direct both implementing the organisation and implementing information supply. Until recently, DEMO's way of working has been focused on shaping the actual architecture of organisations, leaving much freedom for the way of interacting with stakeholders. Recent research has tuned DEMO's way of working to specific needs, e.g. in general organisation design [86], in areas of organisation splitting – to enable Business Process Outsourcing, to use Shared Service Centres in the primary

business or to make (de)centralisation choices [81] – and in the area of application portfolio rationalisation [97].

DYA [148, 147] is an approach to enterprise architecture from Sogeti – Netherlands. DYA stands for DYnamic Architecture and places an emphasis on the process of architecting in general, and more specifically on the development and improvement of the architecture function. DYA was developed from the author's experience that the bottleneck for the core processes of enterprise architecting is not the design of architecture products, but rather the embedding of those products in the organisation. DYA is founded on a number of principles, which can be summarised as:

- The enterprise architecture process is as important as the enterprise architecture products. The objective of an architecture should not be to deliver architecture products, but rather to support the enterprises change processes.
- Just enough, just in time architecture. An enterprise architecture can be implemented step by step, driven by business needs. It is not necessary to design all enterprise architecture products in one step.
- Deviations from the architecture are allowed on occasion, but only in a controlled way.

An essential element of DYA is the DYA model (represented in Figure 5.3). This model contains four processes which cover the enterprise architecture core processes. The Strategic Dialogue (between business and IT management) is concerned with the translation of the business strategy into project proposals. By default projects are realised using the enterprise architecture (Development with architecture), but situations may occur in which projects are realised without conformance to the enterprise architecture (Development without architecture). Within this model, the architecture services processes for defining and managing the architecture are positioned as supporting processes. DYA does not prescribe any methods or techniques on how to develop enterprise architecture products.

As of version 3.0, Capgemini's Integrated Architecture Framework (IAF) [30] makes a distinction between the content of architecture and an architecture process. IAF states that an architecture process should always be tailored to a specific situation, characterised mainly by its leading purpose. For the architecture process, IAF gives, analogous to the alignment perspectives as first introduced in [99] and further elaborated in [55], several "architecture process patterns" called *roadmaps*. IAF gives three examples of such roadmaps, namely (1) integrated top-down route, (2) IT-focused route, and (3) problem-focused route. The *integrated top-down route* roadmap (see Figure 5.4) is mainly used to integrally redefine the structure of business, information, information systems and technology infrastructure, as a means to steer an business/IT transformation. This roadmap is characterised by a top-down pattern, mutual iterative refinement between a broad range of aspect areas and an integrated base-line at the logical level before proceeding to the physical level. In the *IT-focussed route* roadmap (see Figure 5.5) the focus is on understanding the business and information to translate them into architecture for information systems and technical infrastructure. Compared with the integrated top-down route – because of its limited scope – it will be faster, more focused on IT and less complete on the busi-
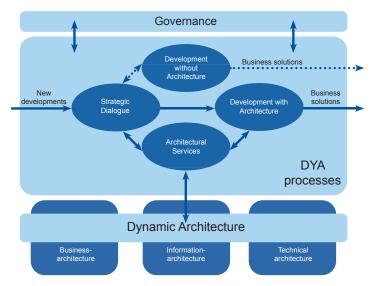
Fig. 5.3 DYA model

ness and information side. Therefore it will typically be useful to support IT-related decisions, accepting the business and information as a given. The *problem-focused route roadmap* (see Figure 5.6) is used to determine the impact and consequences of changes in (the environment of) some current system, so at the physical level. If one understands the consequences at a lower abstraction level and translated them into effects on a higher level, management is able to take measures (bottom-up). These measures at a higher level can then again be translated into principles or guidelines, to be applied at the lower level (top-down).

Applying the IAF-terminology to our focus, the process for enterprise architecture, we could discern a *typical EA-roadmap*, which gives a first elaboration of the overall strategy of the enterprise, clarifying e.g., what and how centrally managed, what are major guiding principles, what needs to be shared and what freedom is left to the lowest levels of the organisation. Such a typical EA roadmap will generally address all aspects areas (Business, Information, Information System and Technical Infrastructure), concepts and services on contextual and conceptual level; principles, guidelines & standards and recommendations for transformation. Where the first two roadmaps can be directly applied to common EA-purposes in supporting decision making, the problem-focused route roadmap can be used to create awareness for enterprise architecture.

IAF focuses on content and the process to create the results. Relatively much freedom and flexibility is given on how to shape and execute the architecture process.
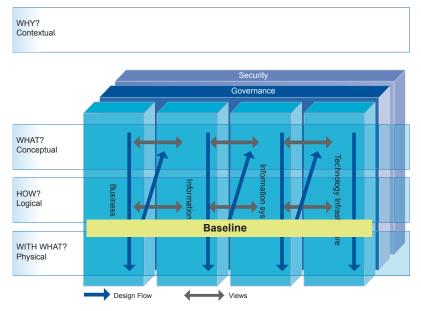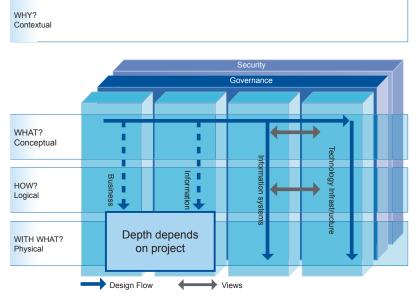
Fig. 5.4 Integrated top-down route roadmap

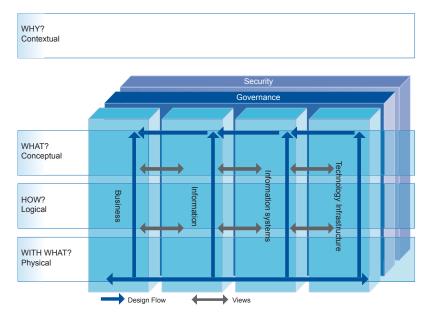**Fig. 5.5** IT-focussed route roadmap

Fig. 5.6 Problem focussed route

5.3.2 Architecture schools: check the "Instruction for use"

As mentioned before, several approaches with a proven track record for the core enterprise architecture processes have emerged, some even taking the form of an architecture school or style. Within each of these schools, we see that the process of creating an enterprise architecture has been elaborated best. The processes involved in for the application and maintenance of an architecture are at best mentioned, but are in need of further elaboration.

We believe that there is a striking resemblance between enterprise architecture schools and approaches to strategy formation. In [85] ten different approaches to strategy formation are described. The authors of [85] argue that each of these approaches or schools have specific contributions and limitations. Some of the schools are prescriptive in nature, concerned with the formulation of a strategy, thus emphasising its content. Other schools describe how a strategy is produced, paying less attention to prescribing the ideal strategic behaviour, thus emphasising its process. As examples of a process school, the *learning school* beliefs that strategy formation is an emergent process in which strategies can only be developed in small steps as an organisation learns over time. Another process school, the *power school* treats strategy formulation as a process of negotiation between conflicting groups within an organisation. A typical example of a content school, the *design school*, states as essence of a good strategy a created fit between internal capabilities and external possibilities, preferably expressed as a Strength-Weakness-Opportunities-Threats (SWOT) analysis.

We see similar characteristics in enterprise architecture schools. Some architecture schools or enterprise architects place emphasis on the aspect of communication with stakeholders. In their view the creation and application of enterprise architecture is only successful if all stakeholders are involved in the process and have expressed their support for the results. Emphasis is put on reaching a shared conceptualisation and less emphasis is put on the quality of the tangible enterprise architecture results themselves. Risks that may arise from these architecture schools are for example: a communication and change process just for the sake of it, the lack of sufficient detail in the results to guide the transformation or even enterprise architecture results that are developed only for shared conceptualisation and not for decision making. Other architecture schools or enterprise architects put an emphasis on tangible enterprise architecture results, often prescribing what method and techniques to apply in developing certain enterprise architecture models. Some potential risks associated with this type of architecture school are: not enough attention for stakeholders, models for the sake of models, too detailed models or focus only on the creation of the enterprise architecture.

Each of these architecture schools have a specific perspective that focuses on one or more specific aspects of enterprise architecture. Each of these perspectives is valuable in itself, providing interesting insight into aspects of enterprise architecture. Each of the architecture school therefore has its value, given that we recognise their difference and underlying premises. Caution, however, must be taken to avoid the pitfall to believe that a single school is apt and applicable to all situations. Note that some have the tendency to promote their architecture school almost as if it is a "religion" to be applied to all situations and that discussing an alternative approach to enterprise architecture might easily lead to an argument.

Before adopting an architecture school, one should assess whether the underlying premises and principles of that school are applicable to the situation at hand. Ideally, every architecture school should contain an instruction for use, explicitly indicating those premises and principles, the intended value, the situations in which (not) to use, risks and the requirements on the enterprise architect and the organisation. Unfortunately, these instructions for use are hardly documented.

5.4 Architecture effectiveness and organisational context

At the beginning of this Chapter, we described the core enterprise architecting processes *create*, *apply* and *maintain*. Not every organisation, which utilises architecture as a steering instrument, should put equal emphasis on each individual architecting process. By recognising architecture maturity levels, organisations are able to position their own architecture efforts and are able to define their architecture process needed. In this Section we discuss an example architecture maturity framework and an example approach for assessing the effectiveness of an organisation's architecture function.

5.4.1 Architecture maturity level of the enterprise

To measure architecture maturity, architecture (capability) maturity models (AMMs or ACMMs) are created. AMMs are based upon capability maturity models [59] that are formal ways to gain control over and improve architecture processes as well as to assess organisation's development competence. Several AMMs exist, for instance the USA Department of Commerce (DoC) ACMM [33] which provides a framework that represents the key components of a productive (IT) architecture process, models that are linked to the Balanced Score Card [71] concept [51]and models for extended-enterprise-architects [120]. All these models have five or six levels of maturity that vary from *initial* to *optimised*.

In this book we use the DoC ACMM as a basis for an AMM since it is widely accepted in the market and it is, in our opinion, not limited to IT architecture. The DoC ACMM contains six maturity levels (see Figure 5.7). Each level has its specific characteristics:

Level 0: None – *No architecture program and no architecture to speak of.*
> At this level an organisation does not have an explicit architecture. Most of the times, only an implicit architecture with a small scope exists, covering only few (most of the times only IT related) aspect areas. In the dialogue with stakeholders, one can use a bottom-up approach by communicating about the successes of creating and using the implicit architecture or a top-down approach by communicating the intended value of architecture for the organisation.

Level 1: Initial – *Informal architecture process underway.*
> This level is characterised by ad hoc and localised processes, no unified architecture process across technologies or business processes, little or no adherence to existing standards, limited management team awareness or involvement in the architecture process and no explicit governance of architectural standards.
>
> At this level, there will be only one (or few) stakeholder(s) who will see the added value of architecture and architecture capabilities only exists with a limited number of people. Enterprise architecture is not embedded in the corporate governance processes and the emphasis in the communication to stakeholders should be in terms of the benefits of architecture and making them enthusiastic. The focus will be on the *creating* process of enterprise architecture.

Level 2: Under Development – *Architecture process is under development.*
> At this level, architecture process has developed clear roles and responsibilities, vision, principles, business linkages, baseline, and target architectures are identified and architecture standards exist, but not necessarily linked to to-be architectures. Reference models and standards have been established. There is an explicit linkage to business strategies, management awareness of architecture effort and governance of a few architectural standards and some adherence to existing standards.
>
> At this level architectures exists and the focus lies on *applying* them. Architecture products are being used in individual decisions and embedded in some

decision documents. The focus in the communication should be on making all relevant stakeholders aware on the added value of enterprise architecture.

Level 3: Defined – *Defined architecture including detailed written procedures.*

At this level the architecture is well defined and communicated to staff and business management with operating unit responsibilities. The process is largely followed, gap analysis and migration plan are completed and architecture is integrated with capital planning and investment control. Also the senior management team is aware of and supportive of the enterprise-wide architecture process and most elements of operating unit show acceptance of or are actively participating in the architecture process.

At this level architects will be consulted as content experts for advice, but don't have authority to enforce compliance to the enterprise architecture.

Level 4: Managed – *Managed and measured architecture process.*

At this level the architecture process is part of the culture. Quality metrics associated with the architecture process are captured, architecture documentation is updated on a regular cycle to reflect the updated architecture, senior management team directly involved in the architecture review process, the entire operating unit accepts and actively participates in the architecture process and formal processes for managing variances feed back into the architecture.

At this level the *maintain* EA process has been implemented and the authority of architects has increased: architecture is embedded in overall governance and decision making processes. EA is used to make informed decisions at senior management level in stead of architecture level.

Level 5: Optimising – *Continuous improvement of the architecture process.*

This level is characterised by concerted efforts to optimise and continuously improve architecture process, architecture process metrics are used to optimise and drive business linkages, business is involved in the continuous process improvements of the architecture, senior management involvement in optimising process improvements in architecture development and governance and feedback on architecture process from all operating unit elements is used to drive architecture process improvements.

Depending on the maturity level, the enterprise will be familiar with the usage and benefits of architecture. The higher the level, the higher the acceptance of architecture as a means and less focus on marketing of enterprise architecture or the process of designing an enterprise architecture is necessary. Each level has its specific subjects to manage: on the lower levels the emphasis is on managing architecture awareness within the organisation, architecture skills and architecture processes. On the higher levels the emphasis is on managing the architecture results, participating in transformation steering, informed decision making and continuing improvement of the architecture function within the organisation. At the lower levels the focus will be on the *creating* enterprise architecture process, while at the middle levels the *apply* enterprise architecture process will be introduced followed by the *maintain* enterprise architecture process at the higher levels.
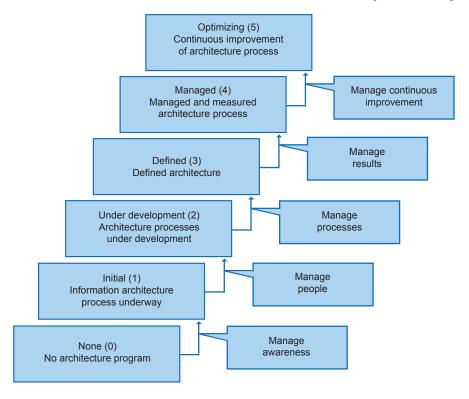
Fig. 5.7 Architecture maturity model

5.4.2 Assessing an organisation's architecture effectiveness

In this Subsection we focus on the effectiveness with which an organisation is engaged in enterprise architecting. The discussion below is strongly based on NAOMI (Normalised Architecture Organisation Maturity Index) as reported in [112]. The NAOMI assessment approach is designed to determine an organisation's architecture effectiveness. It judges an organisation on its ability to reach the goals it set with architecture. NAOMI provides three main assessment variables, which provide three different perspectives on architecture effectiveness:

1. architecture awareness;
2. architecture maturity;
3. architecture alignment.

Enterprise architecture typically starts with an immature level of architecture awareness. The origin of architecture awareness might differ per organisation. It might be the board members initialising an architecture program in order to cope with business complexity, or to guide a large business transformation. On the other

hand, it might also be a few members of the IT department that introduce IT architecture in order to guide software development projects. From this initial, immature, level an organisation should improve its level of architecture awareness. Indications of a high level of architecture awareness are [111]:

- A clear vision of what architecture is, what the objectives are with architecture, and how these objectives are to be realised; the vision, mission and strategy the architecture function has with architecture;
- A clear view on issues such as how architecture should be applied, which aspects of enterprise architecture they see as vital for the success of the architecture program, and how a positive attitude towards architecture is to be created within the affected organisation;
- A clear description of the architecture processes and work instructions.

Architecture maturity indicates how well the architecture function puts architecture into practice. Compared to architecture awareness, this perspective focuses on how well architects succeed in the realisation of architecture instead of how architecture is experienced in the minds of architects. Architecture maturity is about the actual observable behaviour of the architecture function.

An organisation may have several architecture functions, with a focus on different types of architecture. It is key to align those by ensuring the responsible architecture functions cooperate. Such collaboration indicates the level of architecture alignment.

Figure 5.8 shows the structure of NAOMI. A single-headed arrow indicates that the source variable explains the destination variable. A double-headed arrow indicates that the two variables correlate. All six intrinsic variables (Governance, Processes, Communication, Support, Scope, Resources) explain each other, since they all depend on each other. Please note, however, that Figure 5.8 shows only lines from the six intrinsic variables to the three key variables in order to keep the Figure simple.

In order to determine an organisation's levels of architecture awareness, maturity and alignment, NAOMI uses six underlying intrinsic variables:

Governance – This represents the managerial and organisational aspects of enterprise architecture. An architecture function, as well as any other business unit or department, needs to create its own vision, mission and strategy. By doing this, the architecture function states its role and justification of its existence within the organisation, its added value and strengths, its strategic objectives, and the direction in which it wants to reach those objectives. This allows the architecture function to have a clear focus, which should be aligned with the overall corporate (business and/or IT) strategy. Based on its strategic objectives, an architecture needs to create an internal organisational structure, and needs to plan its activities in order to reach those objectives.

Processes – An architecture function should clearly describe its primary and secondary processes. Primary processes involve the development, maintenance, and implementation of architectures. Secondary processes entail architecture
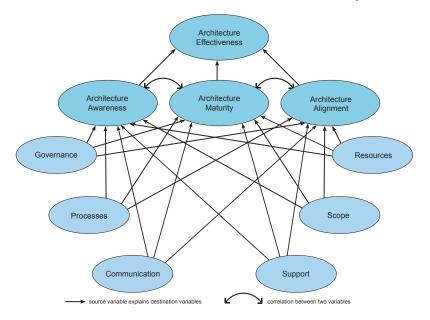
Fig. 5.8 Structure of NAOMI

knowledge and quality management, which focus on improving the quality and
efficiency of the primary architecture processes.

Communication – The level in which an architecture function is able to commu-
nicate with its stakeholders through architecture is essential in determining its
ability to be effective. However, another important issue is how an architecture
function is able to communicate about the architectures it creates.

Support – An architecture function can only successfully put the architectures it
creates into practice with the support of the rest of the organisation. Essential
in determining the level of organisational support for the enterprise architec-
ture program is the level of organisational acceptance of the architecture-driven
changes.

Scope – The organisational scope of architecture indicates which part of an organ-
isation (which departments, units, divisions) is involved in the enterprise archi-
tecture program. The percentage of all departments, business units, or divisions
that work according to architecture indicates the broadness of the architecture
program. The broader the architecture program, the higher the organisational
impact will be. The type of departments – business or IT – where architecture is
being used determines the organisational emphasis of the architecture program.

Resources – The resources an architecture function needs for developing, main-
taining and realising its architectures are twofold. Firstly, it needs human re-
sources – e.g., IT architects, business architects, an architecture manager. Sec-

ondly, it also needs frameworks, methods, techniques and tools, which provide a standardised way of working.

In general one can state that at the lower maturity levels, organisations score low on all NOAMI variables. At the middle maturity levels the score on the variables *processes*, *communication* and *scope* increases while at the highest maturity levels organisations enhance their score on all variables of the NAOMI model.

5.4.3 Assessing an organisation's culture and management style

Organisational culture and *management style* will also be influencing factors on the process. These factors will impact the way how to best cope with stakeholders and how to get them involved in and committed to the process and enterprise architecture results. In an organisation which has a culture of consensus decision-making and therefore many stakeholders which need to be involved, typically the enterprise architecture process will only be successful if a number of workshops with all stakeholders are held to get them involved, to reach consensus and to make decisions. As another example, take an organisation where the most powerful person (e.g. the founder and owner of an organisation) or group (e.g. marketing) decides and others will follow. If this is the case, the process will need to include dedicated activities for communication of the decisions with the others. Organisational culture and management style not only have an influence on the creation of an enterprise architecture, but also on the application and maintenance of an enterprise architecture. As a last example, in an organisation where freedom is valued more than compliance to company rules, using an enterprise architecture to restrict design freedom of individual projects can only be successful if proper attention is given to communication and compliance monitoring.

5.5 Organising the architecture function

We now turn to the question how to get architecture work moving, to keep it working and how to continuously improve it. In Section 5.2 we introduced the creating, applying and maintaining of enterprise architecture as the "core process of enterprise architecting". To get the *act* of this "core process" running, we need to *plan* it and to *learn* from it. We define the *architecture function* as the *PLAN-ACT-LEARN cycle* for the creating, applying and maintaining of enterprise architecture. The PLAN contains the planning of all architecture work (creating, applying and maintaining); it serves to optimally use time and resources of the enterprise architecture team and the stakeholders. In the ACT the core of the architecture work we described in 5.2 is executed according the PLAN. In the LEARN we take lessons from all aspects of the architecture work done, varying from the effectiveness of the enterprise architecture and the quality of architecture results via efficiency of the architecture process to the

causes of all that in organisation, involvement of stakeholders and the competencies and capacity of the architecture team. By a well-implemented LEARN the wheel of continuous improvement to enable the quality of architecture work will get running. The thus defined architecture function needs to be ORGANISEd, i.e. embedded in the organisation with tasks, responsibilities and authority and implemented by people and means. From the PLAN-ACT-LEARN-ORGANISE (PALO), we will now describe the PLAN, the LEARN and the ORGANISE; the ACT has been treated already in Section 5.2.

5.5.1 PLANning activities

By the PLANning activities for enterprise architecture, we understand the activities to continually understand the context of controlled change, the potential contribution of enterprise architecture to the control of this change – so the requirements on the enterprise architecture –, and the approach answering this need, setting the agenda for further activities.

The PLAN contains all preparatory work for the architecture function, in cooperation with its stakeholders, to ensure the architecture work becomes manageable. It typically results in Plans of Approach (PoA) for creating, applying and maintaining, including the way of working (e.g. workshops, interview), the time to be spent and whom to involve. Table 5.1 shows some examples of the planned activities and what the PoA states about these activities.

5.5.2 LEARNing activities

In the LEARNing activities for enterprise architecture (1) lessons are distilled from change processes to further improve the content of the enterprise architecture, the enterprise architecture capabilities and their effects on controlling change, and, based on that, (2) proposals for improvement are concluded, e.g. to more effectively involve stakeholders or to make the next maturity-leap. This learning process is by no means linear; all stakeholders learn, develop and choose while participating in the architecture process. The learning comprises again creating, applying and maintaining. For each of those we will now give examples of learning experiences and improvement proposals:

- LEARN on creating:

  - let's record the updated architecture results in a formal architecture tool;
  - no common framework of reference between the architects is in use; let's adopt a common language for architecture results & process;

- LEARN on applying:

P-activity	Example planned activity	Parts of PoA for example P-activity
P-create	To deliver before the end of the next fiscal year a separate architecture study for the newly acquired organisation, including synergy opportunities with our existing organisation	From the PoA: 5 milestones each culminating in a one-day workshop; enterprise architecture study to be executed with staff from both organisations (own, plus newly acquired); first workshop is the architecture method adoption workshop, to be attended by all architects involved
P-apply	Deliver an application portfolio rationalization proposal to save 30% annual maintenance costs, based on the current architecture models, the formulated business- and IT-strategies	Use questionnaire to build the application inventory and record it in the architecture tool; build business services model & validate this with the business; map application portfolio on business services, detecting gaps and overlaps; let business owners present rationalization proposal to CEO/CFO/CIO
P-maintain	Monitoring relevant external and internal events, resulting in a (1) quarterly newsletter for stakeholders and (2) a yearly one-day-workshop to translate this into a new yearly plan	Each architect may invest 10% of his / her time; Architecture Program Mgt Office produces newsletter; invited stakeholders for yearly workshop involving senior management of all business units, the CIO and CFO

Table 5.1 Example fragments of PLANned architecture activities

- – improve boardroom communication skills;
- – let's design and agree on an exception handling process, when the design team and architecture team cannot reach agreement;
- – our procedure for assigning building permits does not work; reframe the procedure to a more collaborative approach – architecting *by walking around*;

- LEARN on maintaining:

  - – install a classification for defect reporting;
  - – monitoring was so much focused on the business, that technology developments escaped from our attention;
  - – project results haven't consequently been fed-back to the enterprise architecture team; let's assign project budget for that and make it a condition for sign-off.

5.5.3 ORGANISing architecture activities

To get and keep the described cycle of Plan-Act-Learn activities running, the architecture function needs to be ORGANISEd, i.e. embedded in the organisation with tasks, responsibilities and authority and implemented by people and means. Of course this is an example of more generic organisation design, as treated in [7] and [124]. We will therefore restrict ourselves to some typical example results in

organising enterprise architecture activities, namely in assigning responsibilities, the use of certain (architecture process) principles, the use of KPI's and CSF's, the use of out-sourcing/off-shoring, the building of competencies and the use of tools. Generally speaking, we expect that in a more mature organisation also more explicit attention for implementing the architecture function will be given. At a basic level of maturity the way of working will be typically project-bound, while at a higher level of maturity the way of working will be more structural, continuous and embedded in the over-all governance of the enterprise.

A typical result of organising would be a table for Responsible, Accountable, Consulted and Informed roles, the so-called RACI-table. Example expressions from such a RACI-study could be:

- e.g., lead architect reports to CIO council,
- e.g., separate architecture team / let architecture be part of the over-all governance,
- e.g., for establishing technology standards a CTO needs to be consulted,
- e.g., in our architecture team we discern formal roles of solution architect and enterprise architect, from which we expect the following competencies (using the *Competency language* from Chapter 6).

Another typical result of organising enterprise activities would be to produce a list of supported *architecture process principles*, containing e.g.:

- comply or explain,
- independent judgments can only be made by external parties,
- all investments exceeding the EU-tender threshold (2008: €133.000 for deliveries and services) have to be co-signed by the Corporate Architect,
- enterprise architecture we do ourselves, solution architecture is out-sourceable,
- (as suggested by L-maintain) each project should include budget for feeding-back project results into the enterprise architecture; no project sign-off will be given without this feed-back,
- each project is assigned an architect, paid by the corporate architecture budget,
- the sise of the architecture team should be at least 0,2% of the total workforce of the enterprise.

An organised professional architecture function will also state its Key Performance Indicators (KPI's) and its Critical Success Factors (CSF's); examples of that could be:

- KPI-example: customer satisfaction for architecture work should be 7,5 on a scale of 1 to 10 (1=bad, 10=excellent),
- CSF-example: 6% – 10% from the corporate architecture budget should be spent in programmes and projects.

It is important to organise according to the current and aspired maturity-level. A possibility for that is to make this organising a yearly process, profiting from the LEARN-results. Suppose the enterprise is executing a change process from roughly AMM-level 2 to 3, enabling the corporate use of enterprise architecture, impacting

the over-all governance. Then it would be to reserve for this year part of the corporate architecture budget for describing internal, governing and complying processes.

A special question for ORGANISing is the outsourcings question: what part of the architecture work is out-sourceable / off-shorable / shareable to what party. To answer this question, it would help to discern e.g.:

- business architecture, information architecture and ICT architecture work;
- enterprise architecture and solution architecture;
- steering the architecture process and executing the architecture process;
- type of competencies required and available (see Chapter 6);
- type of architecture results needed (see Chapter 4);
- the phase of maturity (AMM), e.g. in AMM-phase 0/1 input by external architects generally will speed up the learning process;
- the degree in which the business processes itself will be outsourced.

In the latter case, enterprise architecture itself will be used to support and monitor compliance between own enterprise processes and outsourced processes/services. Enterprise architecture is prescriptive for the outsourced services, i.e. it is up to the supplier of outsourced services to design and implement their solution within the guidelines of the enterprise architecture, e.g. with respect to reliability or flexibility.

Returning to the outsourcings question for the architecture function in general, we note the following. On one side we see work which should, according to best practice insights, probably stay with the company itself, such as enterprise architecture, business architecture and steering the architecture process. On the other side we see work which is a more likely candidate for outsourcing, e.g., IT architecture, solution architecture and executing the architecture process. And of course this outsourcing question for the architecture function is a specific case for outsourcing strategies in general, as studied by [48, 81].

To a large extent, the success of the enterprise architecture process will be determined by the team that executes the process. In the next Chapter, we will therefore not only discuss the competencies and responsibilities of enterprise architects, but also elaborate on their role in teams [16, 26]. Here we will briefly highlight some factors that apply specifically to *successful enterprise architecture teams* and their impact on the architecture processes:

- An organised team with clearly defined roles and responsibilities, sharing a clear purpose and goal for the architecture work.
- An agreed common language and way of working. This might lead to the selection, adoption and tailoring of an architecture framework, methods, techniques and tools and/or training the team members.
- The right skill sets and competencies (to be discussed in Chapter 6), possibly involving hiring, training and coaching of team members.

The enterprise architecture processes will be more efficient when supported by *tools*. The same, however, holds true for tools as for the process: there is no one-sise-fits-all set of tools that is appropriate for all situations. Tools can range from simple, general purpose office tools to specialised enterprise architecture tools, useful for

developing, applying and maintaining architecture products. At higher levels of architecture maturity, where the continuous use of enterprise architecture is supported, it is worthwhile to implement and use specialised enterprise architecture tools. At lower levels of maturity, the focus should be on people. Do not expect the enterprise architecture tools to do the job.

Other tools to be considered are tools to facilitate communication and decision-making. Group decision support systems for instance can be used to support complex decision making by a large group of people [73, 89, 144, 143, 146]. Using these types of systems, the opinions of (groups of) stakeholders can be systematically collected, ordered, evaluated, discussed and reported. As another example, *business incubators* can speed up the drafting of plans, decision-making and the creating of commitment for all parties involved. It is important to relate the use of these and other tools to the preferred and feasible way of decision-making.

To grow in quality as architecture function, a number of quality frameworks can be applied, such as from Six Sigma [110] and EFQM [54]. We now take as example the EFQM Excellence Model, as described on [38] (see Figure 5.9, taken from [38]):

> *The EFQM Excellence Model is a non-prescriptive framework based on nine criteria. Five of these are 'enablers' and four are 'results'. The 'enabler' criteria cover what an organisation does. The 'results' criteria cover what an organisation achieves. 'Results' are caused by 'enablers' and feedback from 'results' help to improve 'enablers'. The model, which recognises there are many approaches to achieving sustainable excellence in all aspects of performance, is based on the premise that excellent results with respect to performance, customers, people and society are achieved through leadership driving policy and strategy, that is delivered through people partnerships and resources, and processes.*
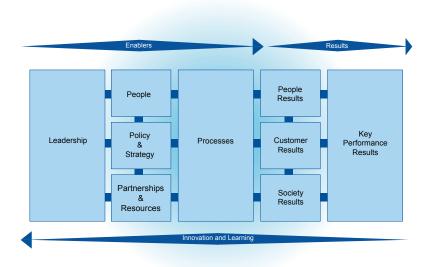

Fig. 5.9 The EFQM Excellence Model

When we compare this with our enterprise work, we see it deals with people (Chapter 6), processes (Chapter 5), results (Chapter 4) and in the end of course effectiveness (Chapter 2 and Chapter 3). In this Chapter, we have also identified the need to learn.

5.6 Summary

As discussed in this Chapter, a good enterprise architecture process depends on a number of situational factors; therefore a one-size-fits-all approach does not exist. It is important to keep in mind that any enterprise architecture is a means to an end; it should deliver value by answering questions of stakeholders. It is essential therefore that the enterprise architecture processes does not solely focus on the delivering the enterprise architecture products, but start from the relevant stakeholders, and understanding their concerns, objectives and stated or implied requirements. In Chapter 4 we already saw that such thinking about purpose, scope, steering needs and expected benefits should determine the architecture results to be delivered. We have seen that this also can influence the architecture process, e.g.,

- to reach agreement or good understanding of scope, purpose, etc., an iterative process might be needed;
- the process around the results will determine the perception of it; even the best results will not work, unless its benefits have been recognised.

Next to this criterion of effectiveness, the architecture process should be efficient, i.e., it should only develop those results (end results and intermediate results) that are necessary to address the concerns of the stakeholders. At the same time the focus in the process should not be on the architecture results, but on the outcomes. An efficient process should therefore include all communication necessary to ensure that results are really applied as intended, including investing time in shaping of relevant views.

Currently, hardly any scientific research or publications exist that describe success factors for enterprise architecture processes. Most of the approaches and architecture maturity models are bundled as best practices of experts. Still, an architecture maturity model can help to cope with architecture maturity and increasing the maturity. The higher level of architecture maturity, the less focus will be on designing and implementing the architecture processes, but more on optimising those processes. This will decide which part of the architecture should get what emphasis; e.g. a valuable policy in phase 0/1 could be "start by a simple result and show to a few IT people". We note that most of those approaches lack an "instruction for use", making the (many times implicit) assumptions on usability and feasibility explicit. Also we note that an architecture process really is a business process as many other processes, which evokes the question how AMM, NAOMI and the drafted PALO-notion could benefit from general quality frameworks such as those from EFQM and Six Sigma.

To common descriptions of the architecture process, which tend to emphasise the creating (order of working and products) and sometimes the applying of architecture results, we added maintaining results. Also we argued that this *Create / Apply / Maintain* activities need not only acting, but also planning and learning. This 3 by 3 matrix *Plan/Act/Learn* versus *Create/Apply/Maintain* needs to be *Organise*d, implemented in the organisation. Also for this Organise, experience is still on the level of typical examples, no patterns do exist yet to combine maturity with each PALO-aspect. For example, it would help to connect maturity levels of the organisation to the role of enterprise architecture and enterprise architects in formal decision making, preventing the use of *building permits* or contracts on AMM-level 0. *Learn* has to look next to *Plan/Act* also to *Organise*: are we still effectively and efficiently organised and are we delivering the results that our key stakeholder value?

We now summarise the core aspects of the process of enterprise architecting.

- Create:

  - Understand purpose and context;
  - Determine deliverables;
  - Monitor context and stakeholders;
  - Create shared conceptualisation;
  - Design creation process;
  - Determine impacts;
  - Communicate.

- Apply:

  - Inform;
  - Support decision-making;
  - Ensure compliance;
  - Make results available;
  - (Re)-communicate.

- Maintain:

  - Monitor context and stakeholders;
  - Assess drivers for change;
  - Update and (re-)communicate.

- Organise:

  - Organise team;
  - Select frameworks, tools and tricks;
  - Communicate about enterprise architecture as a means;
  - Embed enterprise architecting in governance;
  - Monitor maturity;
  - Manage quality;
  - Establish leadership;
  - Innovate.

5.7 Discussion statements

1. Enterprise architecting will only succeed if the architecture is just in time and just enough.
2. With change being a constant, a stable enterprise architecture that will last more than three months, is an illusion. Maintenance of the enterprise architecture will be the core process.
3. There is no one single way of doing enterprise architecture.
4. Each school of architecting has its own virtue.