Dragon1 Fundamentals

Study Guide, 1st Edition

Preparation for the examination "Dragon1 Foundation"

Mark Paauwe

Talitha Paauwe-Wijnands

Dragon1

THE DRAGON1 SOFTWARE COMPANY

Faye, Sapere Aude

Colofon

Website: www.dragon1.org

In addition to this study guide, a of Dragon1 is available to you. On the website www.dragon1.org, you will find open standards, checklists and document templates to get started with Dragon1 in practice.

Title Dragon1 Fundamentals – Study Guide, 1st Edition

Release 17 July 2014

ISBN 978-94-90873-00-4

Authors Mark Paauwe
Talitha Paauwe-Wijnands

Publisher The Dragon1 Software Company

PO Box 239
6700 AE Wageningen
info@dragon1.com
www.dragon1.com

Editor's Note

We have made every effort to trace holders of copyright. Persons or bodies to benefit from certain rights are kindly requested to contact the publisher.

Preface

More and more Enterprise architecture is used in organizations today and also for various purposes.

The government for instance uses enterprise architecture as an instrument to achieve end-to-end-chain-oriented e-services and to achieve a digital workspace. Banking and insurance companies use enterprise architecture often to align their information systems and to support their business processes as optimal as possible.

In the field of education and health care organizations, enterprise architecture enables adaptivity in order to shorten the time to market for new products and services which are tailored to the ever changing needs of the student and the client. Energy utilities, telecommunication, logistics and food companies are using enterprise architecture to increase their production capacity and to increase their quality of service.

However, the use in practice of enterprise architecture in organizations does not fully meet expectations up to now. This causes discussions on the added value of enterprise architecture to arise. Without any doubt it is certain that with architecture beautiful sustainable and future-proof solutions for complex issues can be designed and realized. The big challenge is however how to ensure the benefits of working with enterprise architecture in organizations.

For the Dragon1 Architecture Foundation this is reason enough to support the initiative of a new study guide about Dragon1 and in the same time continue to propagate, disseminate and ensure the correct use of this innovative approach to enterprise architecture.

"A picture says more than 1000 words."
"Documents are in the drawer. Visualization-posters hang on the wall."

Within the open method Dragon1, a new innovative approach has been put foreward. It is a fundamentally different approach, regarding enterprise architecture as a special total concept, focussing on creating a total concept design that answers to the various of different requirements of owner/clients and many stakeholders and visualizing, communicating and realizing the total concept design, based on the approaches similar to building architecture.

In Dragon1, the focus is on creating architecture visualizations tailored for various groups of stakeholders within the enterprise to achieve better perception, judgment and decision making. It is reporting status visually, analyzing and improving the level maturity of business administration (as set of business concepts and information science (a set of information concepts) of the organization, which is mapped on strategy, best practices, norms and standards and made manageable at conceptual, logical and physical level.

A benefit of Dragon1 is that maturity of concepts are measured, strategic issues can be resolved into sustainable, durable and future-proof concepts and solutions that serve enterprises at a strategic level, but will also lead to concepts and solutions that can be implemented at operational level and can be managed properly at tactical level.

As mentioned Dragon1 is a fundamentally different, innovative and new approach. In a period of ten years Dragon1 has been developed and elaborated. In close collaboration with universities, research institutions, governmental institutions and commercial companies, fruitful effort is spend to establish a scientific foundation to support the method.

All parts of which the method consists, are developed and verified in many organizations within different sectors to continuously improve the method. As a result of this a set of Dragon1 open standards and EA best practices are created that together constitute the entire method Dragon1 and its framework with reference models for each common organization type and business function.

Dragon1 is an open method. This means that the method, under certain conditions, is free to use. And it is also possible to suggest or to make improvements and enhancements for inclusion in the method. You can therefore contribute to the further development of Dragon1. The conditions for the application and the manner of submission of amendments can be found at www.dragon1.org.

We wish you, as the reader, much pleasure reading the study guide and applying Dragon1 in practice. Of course we are very much interested in your experiences with Dragon1. For comments and experiences you can contact us via the following email address: info@dragon1.org.

July 2014

Dragon1 Architecture Foundation

Table of Contents

Introduction

Talitha Paauwe-Wijnands

Why a Dragon1 Study Guide?

This Dragon1 study guide is about a new way to manage and control organizations: namely at a strategic level with Visual Enterprise Architecture. People in enterprises and governmental institutions involved in starting, managing and implementing business transformations and innovations are increasingly using Visual Enterprise Architecture as a management control instrument.

This study guide is intended to be used in education, training courses and in practice. Here the future managers and architects are present, who are to shape sustainable and future-proof enterprises and institutions.

What exactly Visual Enterprise Architecture means, is explained in this study guide. For now, it is important to know that Visual Enterprise Architecture is about visualizing and designing the total concept of how an organization works today and in the future in terms of governance, business, information and technology. With visualizations such as sketches, drawings, diagrams and photographic images, of designs of business transformation and projects for IT innovations risk can be controlled much better. This is because visualizations, better than text, make complex issues more decidable.

This study guide lays out Dragon1, the open EA method as approach for Visual Enterprise Architecture. The reader is offered information and theory highly structured and therefore is able to use it better in practice.

Dragon1: the open method of Visual Enterprise Architecture

Dragon1 as a method consists of four 'ways': a way of thinking, a way of working, a way of representing and a way of supporting. In this study guide two ways, the way of thinking and the way of representing, are explained in separate chapters. Before these ways are discussed in detail, in the first chapter the fundamentals of Enterprise Architecture as Total Concepts are explained. The second chapter is an overview of the Dragon1 as Open EA Method. In the third chapter a number of management concepts related to Enterprise Architecture as discussed. Following are the two chapters that further detail the method. This study guide ends with a chapter providing the core of the Glossary of terms of Dragon1.

Dragon1 as a method is based mainly on three disciplines: building architecture, business administration and information science. In fact, we want to achieve here but only one thing: the advantages of building architecture, such as communicative visualizations for sustainable and future-proof structures and solutions, reuse for the design and realization of innovations such as complex and integrated business IT solutions, the structures within enterprises.

Dragon1 positions Visual Enterprise Architecture as a conceptual design activity at the strategic level, which resides between vision and strategy on the one hand and policy development and change management in the other. In programs for transformation and innovation and projects for implementing policy, architecture and architecture designs are used. Visual Enterprise Architecture creates the space and freedom to do business in this way.

In this study guide the different topics discussed in the chapters are summarized in a systematic way. Almost all chapters begin with study objectives. These objectives are attainable at different levels. Therefore, no standard is associated with the learning objectives. The study guide is suitable for a wide range of education and training courses. The answers, additional questions and additional teaching material can be found at www.dragon1.org.

An addition, when we talk about in this study guide about the architect, we mean here the architect who works in the enterprise such as the enterprise architect, business architect, information architect, security architect and technical / IT architect.

Which subjects are explained in this study guide?

Chapter 1. This chapter, 'Visual Enterprise Architecture: a new view on enterprise architecture', offers an alternative conceptualization for architect, manager, CXO or board member. In other words, it offers a new way of working on enterprise architecture, business architecture and information architecture as well as technical/IT architecture as total concepts for domains within an enterprise.

The call for a new view on enterprise architecture originates from practice where the current use of architecture within enterprises often does not fully meet expectations. Therefore in this chapter we take a step back to look at the origins of architecture, how we design, visualize and realize architecture and what architecture really entails.

Chapter 2. This chapter, 'Dragon1 at a glance' provides an overview of the entire method Dragon1. In this chapter, the four method 'ways', most important Dragon1 concepts, and subconcepts are introduced as well as corresponding models and

definitions. The chapter consists of schematics and broad outlines to be read as a summary of the entire Dragon1 method.

Dragon 1 outlines how management and architects can use Dragon1 to organize working with architecture, to design, control, adapt as well as business transformations and innovations within an enterprise.

Chapter 3. This chapter 'Visual Enterprise Architecture in practice' elaborates on how visual enterprise architecture is included within enterprises, as a management control instrument to enable board, directors and management to provide solutions to enterprise issues. Regarding these issues, we distinguish challenges and focal points as well as strategic starting points and principles. We will also discover how visual enterprise architecture can be integrated within an existing quality system and within an enterprise planning & control cycle.

In this chapter we pay attention to a number of common management concepts, such as ambition, strategic intent, concern, strategic starting point, goal, target and activity. We also focus on how management concepts are inter-related with and within visual enterprise architecture.

Chapter 4. This chapter Dragon1 'Way of Thinking' elaborates on the first part of Dragon1. In the 'way of thinking' of Dragon1, the focus is on the theoretical foundations of the method: how does Dragon1 help the architect to observe enterprises and architecture in a different way, in order to display its added value.

The starting point of Dragon1 presents a way of thinking required to enable the architect to design architectures such as a reference architecture, with the ultimate goal to solve challenges within the enterprise. This enables the architect to accept an architecture design assignment to design an enterprise structure, assuring high quality and conforming to performance requirements of the structure.

Chapter 5. This chapter 'Way of Representing' elaborates on the third part of Dragon1. The 'way of representing' focuses on architecture visualization; visualizing the design and realization of total concepts. By using models and presenting stakeholders different views, the architect will work out various perspectives of a structure to show the possible effects and consequences of the requirements issued by the owner/client, stakeholders, such as users. For this purpose the architect has various defined types of architecture visualizations at his disposal, such as architecture visualization-posters, a standard view-layout and visualization design principles.

In the 'way of representing' Dragon1 brings to focus all aspects of defining, visualizing, analyzing and designing structure perspectives, as the architect's principal discipline. It involves getting the architect to bring into view important aspects for which choices need to be made, by means of report views; aspects which require insight and an overview of interdependencies and connectivity, or aspects which require an impact analysis of change.

Chapter 6. This chapter 'Dragon1 Glossary Of Terms' provides an overview of the definitions of the most important architectural terms used in this study guide, consisting of more than 2000 terms of the the open method Dragon1. Because the glossary of terms is substantial, continuously expanding and improving, it is not published entirely in this study guide. The most recent updated version of the complete list of terms part of the open method Dragon1 is published on http://wiki.dragon1.org.

The importance of a common glossary of terms is underlined by everyone, but the glossary of terms has to be sufficiently original, consistent, and mature in order not to cause limitations. These are the conditions of the glossary of terms where innovation is required and Dragon1 provides adequate space.

For the latest information about the contents of this study guide, and the open standards and best practices of Dragon1, please visit www.dragon1.org.

We hope you enjoy studying this Dragon1 Study Guide.

Chapter 1

Visual Enterprise Architecture: a new view on enterprise architecture

Enterprise Architecture has many definitions and shapes.
Be ready to design & realize total concepts for enterprise structures with Dragon1.

1.1 Study Objectives

After studying this chapter the reader will be capable of doing the following:

- to describe how architecture is generally perceived.
- to describe the role of the architect.
- name which building concepts are adopted for the new view on enterprise architecture.
- explain what is meant by a concept, a style element and a principle.
- explain the reasons behind the use of a concept sketch and a principle drawing.
- explain why enterprises can be viewed as structures.
- To name the basic 10 enterprise diagrams and solution diagrams every architect always needs to create in every assignment and for every project.
- To name the two toplevel products an architect creates and the five generic architecture visualizations that are always present in these products.

1.2 Introduction

The chapter, 'A new view on Enterprise Architecture', offers a new school of thought for architects, managers, CXOs and board members in working with enterprise architecture, business architecture and information architecture as well as technical/IT architecture within the enterprise at strategic level to control risks.

The call for a new view on enterprise architecture originates from practice where the current use of architecture within enterprises often does not fully meet expectations. Therefore in this chapter we start off by analyzing the current of enterprise architecture and then take a step back to look at the origins of architecture, how we see and visualize architecture and what architecture really entails.

In this chapter we refer especially to an another discipline, the building architecture. What are the parallels that can be found between enterprise architecture and building architecture? Similarly, can enterprise architects benefit from this? We will discuss the reusability of using some of the architecture concepts used in building architecture in relation to the use of enterprise architecture.

Using this knowledge we take a fresh look at enterprises. We provide a brand new glossary of terms for enterprise architecture and by way of examples, lay the basis for the school of thought for enterprise architecture as total concept for enterprise structures.

1.2.1 Improved business-IT alignment increases Enterprise Adaptivity

To continuously increase the adaptivity of the enterprise, the speed at which an enterprise can react to a everchanging environment, is very important. Adaptivity surpasses money, knowledge and time as most influencial aspect on the survival of the enterprise in current economic and technology times.

Recently enterprise architecture is more and more used as a strategic management control instrument to achieve improved alignment between various businessunits in an enterprise and their supporting means and increased adaptivity of the enterprise as a whole. And also to control risks in business transformation and IT innovation.

Enterprise architecture is often focused on achieving improved alignment between business processes, often seen as the most important building block of the business, and information systems often seen as the most important building of Information Technology. Enterprise Architecture is also focused on using proven technology, best practices, best supplier solutions, reference models, standards and norms in programmes and projects to mitigate risks.

Improved business IT alignment with Enterprise Architrecture is tried to be achieved by first creating and applying various architecture and management products at tactical level such as policies, guiding statement documents (labeled as principles), norms, standards & guidelines documents, best practice based AS-IS and TO-BE reference models and Entity Relationship Diagrams (ERD). Second, in design, change and realization activities, throughout the enterprise, and often projects oriented, architects try to have their created architecture and management products used, in order to realize some kind of standardization, reuse and else.

Unfortunately this approach often does not lead to the desired standardization, reuse or improved business IT alignment and thus also not to the desired increased enterprise adaptivity. So if this way of practicing enterprise architecture does not work, what would then be an approach of enterprise architecture that increases business IT alignment and actually let risk be controlled in business transformation and innovation IT?

In today's world, we recognize a vast number of enterprises delivering products and services to their own internal businesses, to consumers and other businesses. Enterprises operating on a national scale may carry sometimes between five to ten different businesses. In turn these businesses can consist of thousands of different elements (parts of concepts) such as products and services, employees, business processes and activities, information and IT, including information systems, software applications, computers and networks. All these elements need to be finely tuned together in order for individual elements to cooperate optimally with eachother in order to achieve maximum results. In such an enterprise really can be seen as a structure, an **enterprise structure**, having a construction, facilitating operations and decorated to look different from the actual situation for the outside world.

Unfortunately, we often observe unsatisfactory levels of collaboration between the elements in these enterprises and their businesses. When individual elements are

not aligned optimally the whole enterprise suffers and its return on investment will be lower than would have been possible.

With poor alignment of an enterprise's business and IT, we see more information systems, more employees and resources needed to do the same job.

An enterprise's capacity, the work it is expected to carry out with its personnel and resources, can be robustly improved by means of a better business/IT alignment.

After an architecture project we often see that business processes fail to be optimally supported by IT and/or information systems. In order to design an appropriately functional IT or an optimal supporting information system, it is important to understand how a business process works, what the current and future issues are, and how that business process is likely to develop over time.

Often, a business process that needs to be supported by IT is incorrect, incomplete or not visualized, prior to developing the new information system. As a rule this will result in less than optimal business process support. Consequently, as a result of flawed information and wrong assumptions the architecture design, the implementation and the configuration of the information system are created based on the wrong insights.

For instance if it is not fully clear which activities are present in a business process, including the exact name, definition and contents of the activities, chances are that the information system will be designed and built on the basis of an inaccurate functional model and cannot be connected on a 1-to-1 basis to the activities in a business process.

All said in a lot of enterprises there is much need to improve the alignment of all elements in the enterprise structure. There is stil a lot of work to be done there.

1.2.2 The need for a new view on architecture in Government institutions

Aside from enterprises, there are various types of institutions in the non-profit sector, particularly governmental institutions, displaying many common features which they share with enterprises. They also provide products and services to

consumers, individuals, and to businesses and they have their own internal business processes and information.

From a decentralization and innovation point of view, there is an increasing number of local authorities that need to work with the central government on eGovernment / digitalization, in order to see citizens and their internal businesses as customers. Government institutions are thus being set up as businesses. It is therefore important for government institutions such as municipalities and provincial authorities to have a different way of looking at enterprise architecture. This has not been the case up until now. Citizens and internal companies in need of customer services require an entirely new set of concepts and principles.

Central Governments have developed their own reference architectures to provide enterprise architecture with a solid base. This assists municipal and provincal authorities to commence with the implementation of enterprise architecture. In turn, this assists to attain a firmer hold on institutional business and IT, and apply the required overall innovation needed. Municipalities and provincial authorities still need to possess a lot of knowledge about enterprise architecture because reference architecture is not a panacea neither does it provide tailor made solutions.

Fortunately, governmental institutions have begun to improve their services at their own initiative such as working on improved information facility and business operation.

Example of a Dragon1 compliant Architecture Case to Improve Citizen Services Communication at a Municipality

An enterprise architect was assigned to come up with a design proposal for a solution that allowed an municipality to provide their citizins with progress information and status update about serviced delivery and at the same not being hindered to fundamentally restructure the organization.

The architect started to create a enterprise domain diagram and to draw an enterprise function diagram with the domain diagram as background based on input from interviews. The diagrams show in a high level way the usage and processing of citizen applications autonomous from department, employees and workplaces. Also it is visible what status update and other information is communicated to citizens about the progress of their

application.

Next the enterprise architect drew services into the function diagram and interviewed the stakeholders pointed out by the owner-client. This resulted in a large set of stakeholder need, quality and performance requirements for the business and IT services.

With this input the architect created a stakeholder onion diagram, a needs diagram and requirements diagram, showing the relationship of stakeholders, needs, requirements, functions and services. The enterprise architect together with the owner/client analyzed the costs per requirement and using M.O.S.C.O.W. to make a budgeted selection in a capability map and costdiagram.

Next the enterprise architect researched and select the best available and affordable business and IT concepts and including business and IT elements to answer to the requirements for the services. With a solution concepts diagram (holding best practices) and solution elements (holding industry norms and standards) diagram he proposed to the owner-client when and what elements to buy, build and implement for what cost, quality, performance and functionality.

The Owner-Client approved the proposal of the architect and a project was started to realize the architectural solution that was designed. The architect created a 100 pgs architecture description document contain all diagrams and texts and a 20 pgs A3-sized glossy architecture design book in which he put nice artist impressions, structure visions, landscapes, blueprints and scenario roadmaps of the diagrams he had for the enterprise, the solution and mapping of the TO-BE solution onto the AS-IS enterprise.

In the project the visualizations from the design book of the architect were used by all different stakeholders as high level guiding framework for the work to be done. A customer service center and cloud-based case management tool were actually the central elements of the solution designed allowing the organization to restructure departments while keeping the ´shop´ open. During the project the architect and owner/client solved issues by using the formal diagrams to argument, legitimize and support their decisions.

In this example, the enterprise architect separated the demand for logical functions (What) and the need for technical forms (How) in a solution at analyses and design time. And he separated and project the solution from and on the enterprise. He did all this in order to realize an integral solution and implement it in an enterprise structure. He separated requirements, constraints and strategic starting points from

the enterprise functions and the technical solutions elements and achieved a design of a sustainable and future-proof solution.

Note: A governmental institution or a non-profit organization do not have profit making goals, but a social responsibility which makes it significantly different from an enterprise. Knowing that, their architectures are intrinsically different from commercial enterprises because, different principles apply to 'for-profit' as opposed to 'non-profit' organizations. The future will show many institutional architectures to increasingly incorporate 'for-profit' principles.

Until otherwise stated in this book the term 'institution' is replaced by the term 'enterprise.'

The enterprise architect is able to visualize the current AS-IS situation of an enterprise, and to visualize the architecture of an future-proof solution. With these visualizations the client is able to make decisions for the realization of a solution and a fundamental change in the enterprise.

In the two previous sections we outlined in short that enterprises, government institutions and for-profit organizations already are working with architecture, but that the promised results are not yet achieved. In this book we want to make a number of suggestions to reverse this.

The following sections will now first describe the origins of architecture, to get a better understanding of the role and place of architecture and how promised results are achieved when designing and realizing building structures. We explain how architecture is perceived by us (humans) and what architecture entails. We will also explain why it is interesting and also fair to compare the discipline of enterprise architecture with the discipline of building architecture. How do we get as good a match and cooperation as possible between both architecture disciplines?

1.3 What is architecture?

As enterprise architect it is important to do proper evaluation of the mechanics of systems - the way they work – in general. The better you observe, analyze and understand the mechanics of a system, the more you are able to reconstruct the mechanics of the system elsewhere or apply the knowledge you gained in other added value way. It is an architects task to design and realize new systems that function optimally. In order to be able to do this effectively a good knowledge of how the world works is of most significant importance.

1.3.1 Observing architecture, concepts, elements and principles

Humans principally use their eyes (sight) and hands (touch) when observing objects. Seeing and feeling is the starting point of observation. In the first instance we gather information about our surroundings by looking at it and touching it. This, of course, does not mean that the other senses do not play a role in the final perception. When we walk through a mall, we will view the mall differently than other pedestrians, who are also present. We attribute much value, for instance, to greenery in a street, and attribute value to a well-maintained sidewalk (adjacent to the ground floor of a structure). While others could attribute more value to well-designed benches, and trash cans and the presence of sufficient parking facilities.

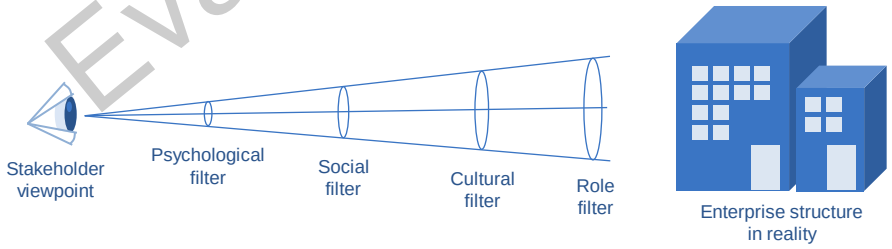

Figure 1.1. The process of perception.

The process of perception consists of several phases. In the first phase we gather information about our surroundings by looking and feeling. In the next phase, the cognitive phase, we structure the information with our knowledge and previous experiences. Herewith we create an image of our surroundings that appears to be

logical to us. At this point, four different filters play an important role, the psychological, the social, the cultural and the role filter.

The psychological filter is made up from a person's personal preferences, the thoughts and emotional well being of that person at the moment of observation. The social filter pertains to social circumstances such as a social group or social class to which a person belongs. The cultural filter is determined by the culture in which a person grew up and the language spoken by that person.

The rol filter is determined by the function that a person performs, manager versus employee, or place in a group, leader versus follower. For architecture, the rol filter, in case of decision supporting visualizations to clients, is of great importance. The four filters are sensitive to observations that are part of the person. Other information is thus filtered out.

These four filters are sensitive to observations that are part of a person. Other information is filtered out.

We design this structured and personal image in the last phase of an evaluation. Added to this, we give a value to our image of the surroundings. For instance, we conclude that we appreciate a mall or find it too busy. For each new observation, the image of the previous observation is further enhanced. As such a cyclical observation process is created whereby the initial observation details are influenced by new observations.

When we are looking and feeling we observe how things, the elements of concepts, act and how things work, sometimes wondering, sometimes understanding, sometimes knowing and even sometimes changing the way things work. We are or become aware of principles. And all these filters play a role in what we think we see and how we see them and value them. And to be aware of this is very important as architect, because it influences what we can or may see and use as (directions of) solutions when designing architecture (total concepts). It also ensures that every architect comes with another beste solution for the same problem.

1.3.2 Architecture is everywhere

We can state that every structure, i.e. everything that has a construction and operations, has an architecture. This applies to every conceivable system, such as a

house, an enterprise, a human being, the world. The architecture of a house or enterprise does not necessarily have to pass a general consensus or be successful or consistent, but the correct or best architecture for the owner-client of a house or an enterprise often takes into account durability and sustainability.

The actual final architecture of a structure is most often an indirect result of the design (exceptions) and realization (alternatives) of a structure. A structure designed in accordance with architecture principles signifies itself by a harmonious entity of constructive, operative and decorative functions. Hereby solutions were found to cater for requirements and wishes which appeared to be opposing and irreconcilable. The ultimate challenge for every architect.

To design a structure that looks like a cube and is positioned out of balance challenges our powers of observation regarding something that is meant to be habitable, but does not comply to the archetype. This is good example of architectural art.

- *Architecture, as field of work, is the art and science of planned, function-oriented and integrated design and realization of structures that are often to be durable and future-proof.*
- *Architecture, in the sense of architecture of a structure, is a specialized total concept. Architecture is the coherent, not neccesarily consistent, whole of decorative, operative and constructive concepts that is or will be applied onto a structure.*
- *Art is 'an irrational and lateral way in which something is done'. Art places the emphasis on the decorative functionality to create a certain atmosphere, emotion, feeling and experience.*
- *Science is 'a structured, systematic way in which something is done'. Science emphasizes the constructive functionality, operational functionality and technical format, creating a certain tensile strength, stability and operability.*

As a rule structural quality increases when it is designed and realized by architecture principles. A robust architecture is an architecture whereby ingenious solutions are found for complex design issues. If this is not the case then we consider the architecture to be weak.

The architecture of the Erasmus Bridge, in the port of Rotterdam, is a great example of a robust architecture. The Erasmus bridge incorporates constructive cable support concepts in order for the bridge to withstand wind and weather conditions.

Functional traffic flow concepts are used that assure cars and trucks can drive across the bridge uninhibited and with speed. A robust architecture sometimes elevates itself to a much used architecture application and therewith an architecture style in which one or more style elements frequently reappear. The Gothic church architecture, for instance, utilizes a pointed arch concept, which is used for construction but doubles up as a style for decoration.

- *Decoration is a non-operative or non-constructive cladding of a structure utilizing elements or components.*

The architecture of a structure is the appropriate architecture if it contributes to the implementation of the client's intended strategy. For instance, the architecture of the Erasmus Bridge contributes to the image of Rotterdam as a maritime industrial city. The bridge constitutes a seamless and eye-catching steel construction link as a result of the use of decorative glossy-steel concepts, exactly as the client demanded.

Note: In building architecture, building is often described as: 'totally or partially install, erect, revamp, modify or enlarge a structure.' Also, it applies to: 'total or partial erection, revamp, refurbishing or changing and enlarging a structure site.' When we consider a construction or a structure a building permit is required, by which construction activities and design of a structure are tested against, among others, a building destination plan.

The crux of the matter is that changes, adaptations and other far reaching activities to a structure or the surrounding area are of sufficient quality when they avoid unceccesary disturbances.

- *A structure is every construction of a physical material such as for instance wood, or digital material such as automated process support, more often than not, furnished for operational or decorative purposes. The automated process support at its end destination is directly or indirectly supported on the basis of, a platform where, services or functionality are delivered locally or remotely. A structure consists of constructive, operative and decorative concepts, elements, components, objects and products.*

By looking at structures such as bridges, structures such as churches, landscapes such as gardens and enterprises such as banks, insurance companies, government and health care institutions, we can learn much from the approaches taken by building architects. Building architecture and enterprise architecture have the same

goal in common, they try to achieve the: qualitative design and realization of sustainable, future-proof, and beautiful structures.

Note: A distinct difference between building architecture and construction is that architecture mostly adds aesthetical aspects to the construction of a structure. Architecture = Construction + Construction Art.

For instance, the Erasmus Bridge (an architectural solution). A chosen architectural solution is often considered heavier than strictly necessary from a constructive point of view, offers operationally more than is necessary and is more pleasing to the eye than strictly necessary compared to a purely structural solution. An architecture solution is therefore often more expensive in the short term than a purely technical structural solution.

We gave an example of architect but you as reader of this book we would like to ask to look around and see or feel what architecture, concepts, elements and principles are around you? And do you see the same architecture, concepts, elements and principles as your collegues, friends and family members? Probably not.

1.4 Building architecture as a basis for Dragon1

Considering the success building architects have with their designs and realization of beautiful, sustainable and future-proof structures it is sufficient reason for us to take a closer look at building architecture approaches; to learn how the enterprise architect can realize sustainable and future-proof structures composed of business processes, information systems and IT infrastructures.

1.4.1 The current practice of a building architect

A profession that explicitly deals with objects that are looked and used by people is the profession of the building architect. Building architects are involved in designing and constructing physical structures. Constructors and builders are actually responsible for the building (realization) of the building project. The building architect is often seen in a technical and administrative role working on behalf of a client regarding the realization of the structure.

Before a structure is being built by a contractor, a building architect communicates with his client on the basis of a portfolio of design sketches and drawings, to illustrate what the structure will look like and will function.

Another professional group, who since 1990 has been involved in modeling and realizing transparent processes and systems in enterprises, is the group of IT architects. This group of architects is involved in developing and realizing solutions to improve the business and IT alignment and to raise the quality of processes and systems.

Certain parallels can be drawn between these two professional architecture groups, and therefore there is a reason for a more detailed look into the discipline of building architecture.

By investigating how a building architecture is prepared and used for the design and realization of an object, we are able to compare this with the activities that enterprise architects are currently doing concerning their architecture. We can thus determine if and how the enterprise architects could benefit from the discipline of building architecture.

The comparison of the two architectural disciplines is as follows:

Building architecture	Enterprise Architecture
The building architect is continuously looking for a design assignment in which he can apply his building construction vision. In order to obtain an assignment he often needs to compete with other architects to qualify.	An enterprise architect awaits a request to design an architecture. He usually does not compete with others to qualify to obtain a design contract.
The architect is the designer of the total concept. He creates the conceptual translation of the customer's strategic wishes to arrive at the required solution by way of concepts and principles. The architect of a church structure mainly concentrates on the quality of the construction of the arches, windows, walls and buttresses. He is less concerned with the construction of the organ, the pews and the amplifier of the sound system, however, he might be involved in the selection thereof.	The enterprise architect creates the business processes architecture and the information systems architecture. The enterprise architect is mainly engaged in designing the enterprise's businesses and business functions based on guiding statements. He also streamlines the enterprise value chains and improves the business IT alignment.
A structure is inanimate, but always used by people. People, who use the structures, are usually not the same people who initiate changes to the structure construction itself. Their activities are however concerned with the interior design of the structure and the decoration. The inanimate parts of a structure contains features that the architect needs to take into consideration, for instance that the entire structure exhibits predictable	An enterprise is a living thing, because it consists of people working together. Supported by resources, to reach several goals as efficiently as possible. Besides design and decoration, there are people within their enterprises who are constantly developing new requirements and changing the enterprise construction. Under a certain pressure, people cause an enterprise's behavior to become unpredictable

behavior patterns under a given load.	as a whole.
Most changes to a structure do not make that structure necessarily more complex. Due to the pursued quality, principles and rules that need to be adhered to.	Any change to (or transformation of) an enterprise makes that enterprise as a whole often more complicated. Insufficient effort to achieve certain quality aspects, such adaptivity. The focus is too often aimed on time, money and the technical structure of systems.
There are documented reference architectures for most types of structure. There are legal and regulatory requirements that need to be met in the design, modification and building a certain type of structure.	For most types of enterprises, no documented reference architectures exist. There are insufficient legal regulations and there are no regulations or rules to be followed during the design, change and achieving a certain type of enterprise solution or part thereof.

Figure 1.2. Differences in architecture practices.

From Figure 1.2 we can conclude that building architecture is primarily a visual design discipline to achieve a better quality in building structures. This is achieved by means of architecture concepts and principles to create a design. We can also conclude that the current practice of enterprise architecture does not use any building architecture practices.

Enterprise architects in enterprises focus more on the modeling and description of architectures and on the the mapping of structures, than on creating architecture designs of enterprises or large parts thereof.

Enterprise architects must actively pursue an assignment to achieve a more predictable outcome rather than just taking a passive directional role. For instance, by having a design assignment for the design of an entirely new innovative enterprise, an optimal information facility or a virtualized IT infrastructure.

From Figure 1.2 we can also conclude that the question asked of building architecture may be the same as the enterprise architecture, which is, 'how can we increase the quality of processes and systems of structures?'

1.4.2 The building perspective: the principle-based approach

Since the beginning of time people in building architecture work with the approach that a structure is a system consistsing of hundreds or sometimes thousands of interconnected components. These components are interrelated and affect each other according to well defined principles and rules. This principle-based approach ensures that the behavior of the structure is understood and controlled as a complete entity as well as understanding and controlling the individual components.

The Erasmus Bridge in Rotterdam is an example of such a principle-based approach. It allows wind and weather to pass through by means of its angled tower and many suspension cables that together form an elegant though robust total structure. A rule that applies to this system is the bridge cable system where the cables are directly fastened to the deck of the bridge.

In the beginning, when the bridge had just been finished it seemed that the bridge was 'swinging', or rather, that resonance was produced by the bridge cables. This phenomenon was caused by raindrops, which caused the air resistance of the suspension cables to change so that the bridge was more susceptible to wind. In order to subdue the effect of wind tremor, new dampers were directly connected to the deck of the bridge and the bridge does not swing any longer.

Another example of a principle-based approach is the building of houses according to 'Huf Haus' architecture. The 'green®evolution' principle ensures that houses have well insulated walls, called constructional insulation. These walls are made of wooden structures. Not only the walls, but also other components such as doors, windows, floors and roofs all form an entity that provides a pleasant living environment.

- *A system is a set of components that work together to achieve a common goal.*

Besides structures, there are other types of systems such as concepts and phenomena. In these, concepts can be considered to be abstractions or ideas by which certain phenomena can be observed. These three types of systems: structures, concepts and phenomena, are explained and defined in further detail in chapter four.

The constituent parts of a system, we call entities. There are different types of entities such as concepts, elements, components, objects and products. These entities in turn can also be regarded as systems. When we consider a concept as a system, then the entities of the concept itself are also regarded as concepts, elements, components, objects and products.

- *An entity is something that is recognized or acknowledged. An entity has its own identity and can be distinguished from other entities. An entity has attributes that provide the entity with an identity.*

An example of a concept of a new housing development is sustainable living. All elements, components, objects and products are assembled to enable the concept to work.

- *A concept is an approach, way of working, an abstraction of a solution, or an idea. A concept is nothing more than a direction for a solution / solution course.*

An example of an element of a house is a facade. An element is sometimes inseparable from a structure. Then we call it a supporting element, or a pillar. It cannot be removed from a structure without that whole structure collapsing and losing its coherence.

- *An element is a more generic, logical functional (composite) component of a system. An element is a part of a systems, such as a concept, a structure or phenomenon.*

There is a difference between elements and components, allowing the building architect to design solutions separate from the technical form. This provides the architect with more possibilities to join opposing requirements of client and stakeholder, and allow more freedom in the design and realization of sustainable and future-proof solutions.

- *A component is a more specific technical physical (composite) entity of a system, with it's own characteristics, properties and/or behavior. A component is a part of a system such as a concept, a structure or a phenomenon.*

Daylight enhancing windows and attic stairs are examples of commonly used elements in houses, creating components which give meaning to a function. For instance, components such as aluminium sliding windows and an FSC wooden

spiral staircase could be included by building architects in the design of houses in a new housing estate.

- *A function is the execution of tasks by an entity. A function dictates what an entity is able to do. A function is the total of the results that can be provided by an element or component. A function does not have a physical form or implementation form, a function does not yet possess a physical identify or implementation identity – a form, however always has one or more functions.*
- *A (technical) form is the digital or physical expression or manifestation of the total. A technical form embodies aspects of implementation. A form is often depicted by an object or component.*

In addition to components we also recognize objects on a physical and digital level. There is relatively little difference between objects and components. A distinction that can often be made is that objects can be entities from which one can produce a collection of information, that can be used for workflow or processing tasks of data.

- *An object is a more specific technical abstract (composite) component of a system characteristics, properties and/or behavior. An object is thus a part of a structure, concept or phenomenon.*

Components, objects and products can also be seen as artifacts. Artifacts are often beautiful, but generally cost too much time and money to use, maintain and improve. The Eiffel Tower and a Chinese Ming vase are precious artifacts from the physical world.

- *An artifact is (often an unique) tool, structure, element, component, object or product recognized by people, which is designed and made with great skill.*

Certain combinations of entities which constitute a (reusable) finished whole (form) are building blocks. Building blocks can be very small, as well as very large. The boundary for determining the optimal size of building blocks is not absolute, so to say, but a building architect always does this explicitly in order to determine the building blocks that are implicitly present.

Note: In between structure and building block is the term loose structure. This is actually a stand-alone structure, a rather mobile structure or a structure that is not embedded in its surface. Loose structure has an inferior sound like 'just a construction' and is therefore not widely used in practice.

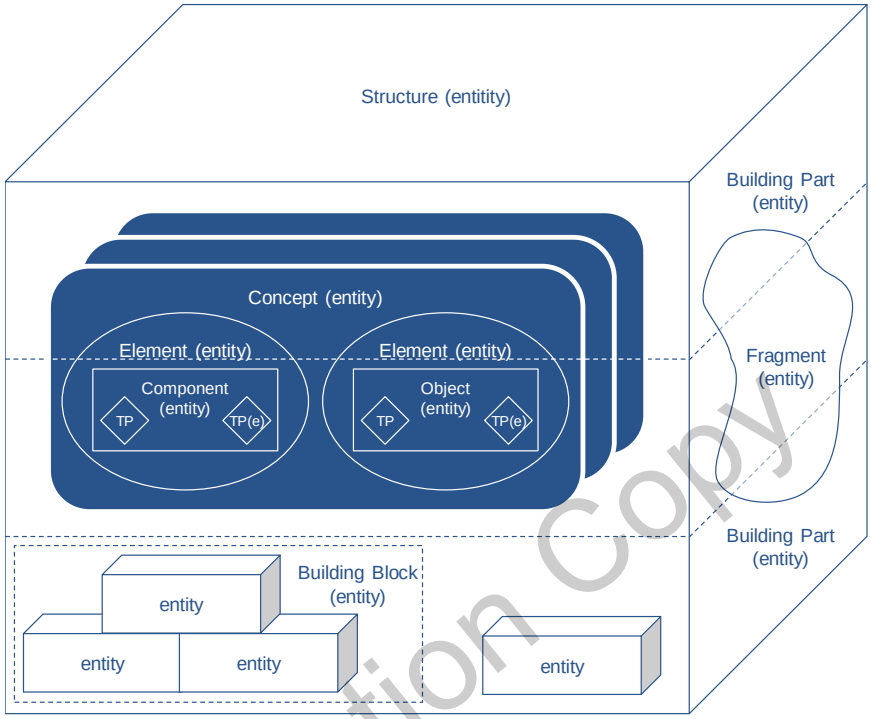

Figure 1.3. Example of a structure with its entities.

Modular construction of homes is an architecture style wherein parts of the house can be put together into a building block at an assembly location. At the final building site, houses are made by connecting building blocks. Looking at the architecture of these modular homes we can see that certain building blocks can be reused again and again. The advantage of building blocks is that certain assembly operations require special knowledge or tools that are site-specific, or they can be more affordable if these work activities can take place on external building sites.

- *A building block consists of a total of entities meant to be used again (repetitively) in a structure.*

There where building blocks represent a total entity, fragments are incomplete parts of a total entity. A building architect will sometimes emphasize fragments of an architecture or a design, or will remodel these because they comply with the solution to a difficult issue in terms of scope.

It's good to know whether the building architect has something to do with a fragment or a building block. By analyzing fragments and determining building blocks an architect can divide complex structures into divisible and manageable work packages.

A striking example of a fragment is a wall of a ruin that is still standing. If the structure is then to be re-instated, the fragment could be a model for the entire reconstruction of the structure.

- *A fragment is an incomplete section that is part of a structure.*

In architecture of patterns are used, to create repetition and standardize they way a design issue, problem or requirement is solved. Architects are designers of total concepts and are confronted with a complex set of requirements. A library of patterns for one thing helps the architect to solve a large number of common problems or design issue quickly and correctly. Patterns are much alike concepts. One could say that all patterns are concepts but not all concepts are patterns and that almost every concept has at least one pattern. Often it comes down to assigning certain tasks and responsibilities to collaborating entities together forming the pattern. In an enterprise as structure one could identify patterns in every business like: the Front-Mid-Back-office pattern. This pattern solves in a generic way what functions to put where in the organization. In the front office customer intimacy and making exceptions may be needed to excel so all client contact is done there. In the back office operational excellence and standardization may be needed so all processing of data mutations and transactions is put in the backoffice, not disturbing the front office activities and vice versa. Other common enterprise patterns are:

- *A pattern is an arrangement of entities meant to solve a problem in a generic way. The problem solved is not part of the pattern itself. A pattern is a specialized concept.*

Because a concept always has entities in some form of arrangement all patterns are concepts. If the arrangement is unclear, unknown or there is only one entitie, or elements and components are unknown, the concept is often used as and called concept instead of being named or used as pattern. The name and symbol of a pattern often reveals the important collaborarting patterns.

Dragon1 has a default set of concepts and patterns that help the architect to get on his way and extend the Dragon1 library of concepts, patterns, principles and elements.

There are always rules which apply to systematic entities. For example at an intersection with traffic lights the rule of a green traffic light frequently applies to traffic only, which is not allowed to cross the other's path at the same time. Another example of a rule that is applied by building architects to houses is that every room in the house has its own function and therefore its own associated form.

- *A rule is an agreement concerning the relationship between two or more entities. If the agreement is not abided by, a sanction comes into force consisting of a variety of consequences. The severity of the sanctional consequences determines whether it becomes a rule, a regulation, a direction or a law.*

If builders perceive a rule as a direction, because the breach of it results in a light penalty, then the client or building architect can increase the severity of the rule for it to become a regulation. In this case the sanction becomes more communicative, resulting in increased supervision regarding compliance of directives and rules.

- *A directive is a rule that imposes a low sanction because non compliance does not have large consequences. Non compliance of a directive therefore causes few consequences for the parties who agreed the directive.*

An example of a regulation applying to each building project would be: 'only structures tested and approved on the basis of legal standards may be included in a building project.' This regulation does not say that no constructions should be developed that are not tested against standards. The regulation does state, however that these constructions cannot automatically be allowed into production. The challenge for many building projects is governing such regulations.

- *A regulation is often a non-binding or mandatory rule brought forth from the branch of architecture. The penalty for failing to follow a regulation is generally non approval of a structure for delivery.*

An example of the consequence a building law is that when building mistakes are deliberately covered up, the responsible employee will be dismissed with immediate effect. In building projects one could incorporate a law, which states that discovery of

a non-documented and therefore unmanageable solution to the implementation of a structure will be removed from the building project forthwith.

A person can also put pressure to bear on the builders who do not report or who ignore, or who omit to document unmanageable implementation solutions, these will incurr an official warning. Such a law could increase the quality of structures, but because of the current situation with building projects this could incurr considerable resistance.

- *A law is often a binding or mandatory rule that is issued by regulators and legislators, which will result in a severe sanction for non compliance. Enforcement of a law tries to prevent the breach thereof.*

1.5 Conceptualization and Visualization

The world around us is full of concepts that underpin the structures that we observe. A structure often consists of dozens of seemingly contradictory incompatible concepts that make the structure an unique complete entity. Building architects who design structures are well aware of concepts, style elements and principles not only to make a constructive and functional entity but also to create an appealing eye-catching building.

1.5.1 Designing from a concept

From the drawing board, thus before a structure is actually implemented, a building architect uses concepts, principles and style elements to predict as far as possible how the components of a structure will interact, respond, support, and react under normal use and under extreme stress.

When we look again at the Erasmus Bridge, then we find this to be a unique bridge because of its angled main pylon (tower), although it is an accepted standard type of bridge as it is a type of cable bridge. The cable bridge is a typical kind of bridge, such as a drawbridge or a suspension bridge. Its' architect, Ben van Berkel, designed the Erasmus Bridge in 1989 from a going concept– the Erasmus bridge measures 139 metres in height, 34 metres in width, has a length of 802 metres whereby the cable bridge section measures 410 metres long. The principle concept of the cable bridge allowed the concept to be used for a maximum span of 808 metres.

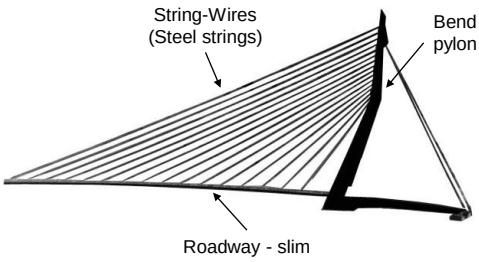

Figure 1.4. Drawing of the concept used for building the Erasmus Bridge.

The Erasmus Bridge is as such designed on the basis of a concept of a cable bridge with pylons as style elements and the principle of 'allowing energy to disperse via pillars into the earth'.

When we look at a house built according to 'Huf Haus' architecture, we see a house that is very energy efficient, made of natural materials and with many recurring constructive style elements. In the Huf House one notices the recurring principle of constructive insulation. The building architects have achieved this by working with concepts, style elements and principles.

A building architect is contracted by clients to create architecture designs. The architect is thereby given a set of terms and conditions. To create these designs, the architect must resolve design issues and make design decisions such as: 'What will the final result look like? What should be the quality of the structure? How strong should I make this structure considering the normal and extensive use of the structure? What appeals to the client and users?'

The building architect must reconcile seemingly incompatible and contradictory requirements in the design, for instance: 'design a large structure, in which one does not feel lost.' For these design issues to be resolved, a building architect uses concepts and designs as such the total concept.

A total concept could consist of many subconcepts. Concepts still allow a lot of freedom regarding potential choices. Nothing is final. Sustainable living, energy-saving constructions and cradle-to-cradle (the concept of the utilization of recycling waste materials) are examples of the total concepts from building architecture that again share many concept components.

On one hand a concept gives a building architect a grip, because the concept includes, for example concept components with proven effect and proven solutions. On the other hand a concept provides a building architect with the ability to communicate with the client about mindset, appropriate solutions and about the desired end result.

Example of conflicting requirements for a villa

An architect was asked to design a house for people who cannot walk upstairs. However, the villa must have three floors with the ability for a person to live independently on each floor. The people do not want to feel like they are living in an apartment complex. The villa apartments must be accessable from the outside by bicycles and wheelchairs. The villa is to stand on a mountainside. The architect combines a winding access road around a three storey villa making it into an integrated total. He utilizes known successful concepts to

create an entirely new concept.

In Figure 1.5 projected on the upper left hand side is a functional requirements model: 'walking, sitting, standing and lying.' Underneath is a technical total concept model projected with partial solutions to allow walking, sitting, standing and lying'. Note: Functional requirements (functions) and technical form (design) in the architecture are analyzed and designed separately.

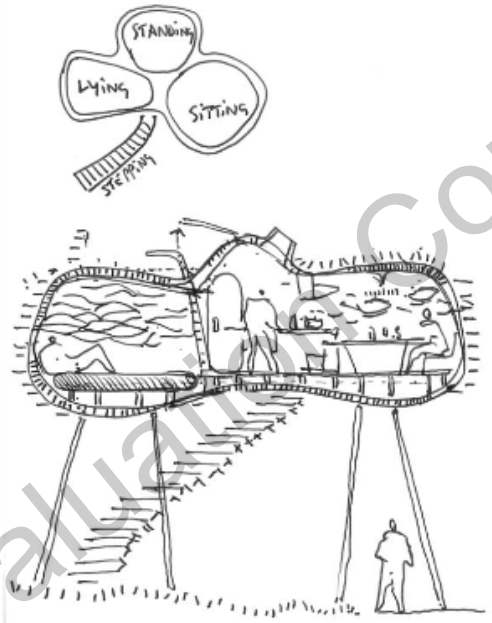

Figure 1.5. Example of a concept sketch of 'living in a dream'.

- *A concept sketch is a graphical image of a concept whereby interested parties, such as the client, in an informal way, are made aware of what the concepts consist of in essence. This includes more or less about how the concept works, what advantages are achieved, and how it might look.*

This is exactly how concepts work. We look at the implementation of a unique physical structure which is translated in our minds into something that we recognize. A concept is an abstract image of reality. It is a depiction of a global or generic direction of a solution for a design issue and is often independent of the actual implementation.

- *Conceptualization is the translation of requirements, starting points and prerequisites by the architect for creation of an architecture design based on concepts, principles, elements and rules to achieve the realization of implementable buildable solutions.*

If we look at a physical structure that forms a connection over water and that gives us the feeling of dealing with a solid and reliable bridge, we will feel confident to use this bridge because we learned from the past that bridges were convenient solutions that proved to be sustainable and future- proof creating a safe passage over water.

That is why many building architects take a bridge as a concept if they need to design a passage over water. However, a tunnel, a ferry, a dam and a helicopter are also alternative examples of concepts that can be created to enable the passage over or underneath water. Each concept has advantages and disadvantages.

In any case one concept suits better than another concept to the client's requirements because it combines all (conflicting) requirements together. Another concept is better suited in the vision of the building architect and anotherone suits better to a municipal plan. That way architects come to the design of, for instance, a bridge, which incorporates style elements of a dam, tunnel and ferry.

A well formulated requirement always contains the word 'must'. For instance, 'a resident **must** always identify himself with a current magnetic identification key, such as a smart card, in order to open the automatic barriers of a driveway.' Such a requirement results in depositing identity information into the security information system of a house.

The building architect takes care that permitted requirements are correctly formulated, prioritized and definitively included for the client.

- *A requirement is something that definitively should come into existence. It is a necessary feature, function or performance to be delivered from something. It is urgently required, as opposed to a wish.*

The building architect will be advised to choose as many fundamental and technological concepts as possible for his architecture and choose as little as possible product or vendor dependent concepts.

Suppose a lighting solution supplier does not require a constant voltage of 220 volts, but uses alternating voltage of between 90 volts and 150 for a unique energy saving form of lighting. The building architect designs in the house for the 90/150 volt alternating voltage. As long as people want to enjoy the financial benefits, they are tied into this solution supplier, because solutions from other suppliers require 220 volts.

1.5.2 Using principles to increase design quality

When designing a structure, the building architect uses principles, called design principles, because they will increase the quality of the design. Principles are statements that relate to how a concept fits together and works, always and everywhere in the same way. Principles describe the way style elements of the concept cooperate and how they realize a certain qualitative advantage.

> *A principle is the enforced way by which an entity operates, happens or takes place with a certain effect or result with regards to a given context.*

The building architect, landscape architect and industrial designer make use of different principles, among others:

- **Principles** to help understand the functioning of the world, behavior of people, the impact and demands of people to visualize and to communicate. For example the principle of 'opposites attract'.

- **Concept principles** that describe how concepts work and whereby knowledge or understanding of the concept principle supports the choice of a concept as a design solution course. For instance, the piston principle of a diesel engine. The diesel engine itself must enforce the operation of the piston.

- **Design principles** specify which maintained operation of an entity should be considered in the design and construction of a structure. This in order to produce an certain result, following an action hold by in an entity. These aspects are in turn present in the context (or surroundings) of a structure yet to be designed and implemented.
  Examples of design principles are: the proportional design principle and mirror asymetrical design principle. People experience the application of the

principle of proportion as pleasant, so the architect must take this into account in the design and construction. For example when designing a garden.
Gravity is also an example of an design principle. Gravity itself is a principle which the architect must always take into account when designing and building an aircraft, church or bridge operation. For example, the Earth's gravitational design principle.

- **Reality Principles** that indicate what principles are implemented in an existing structure or phenomenon, in order to understand and analyze the retained (persistent, mostly undesirable) effects of the current structures and phenomena. Note: These principles recurr in the requirements made on the enforced operation of a structure yet to be implemented.

- **Architecture principles** are especially used to transfer from one dicipline to another and to be reused in another discipline Architecture principles are principles that apply to (almost) all concepts that are part of the architecture (the total concept) of a structure. Architecture principles make clear which integrated operations are enforced in all entities such as concepts, elements, components and objects of a reused structure, or will be applied and which will deliver benefits (or that produces results).

All these types of principles are among others derived from the many books on building architecture, landscape architecture and through conducting interviews with building architects.

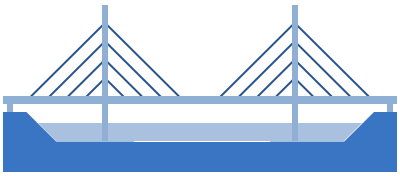

Figure 1.6. Schematic drawing of the cable bridge.

Building architects make use of small visualizations. The principle of a pull cable of a suspension bridge is visualized Figure 1.6. The principle is formulated as follows: '**By** always having suspension in the cables and having pylons to carry the weight of the bridge to disperse the energy to the foundation of the bridge, from steel cables attached directly to the pylons and the bridge deck, **it ensures that** the thickness of

the deck can be kept to a minimum (about 2.5 meters) **whereby** vehicles using the bridge have less of an incline to climb.'

- *A schematic drawing is an informal visual depiction with more detail than a sketch, in order to explain to stakeholders such as clients how a concept, structure or phenomenon works. This drawing is used to support the decision for choosing a solution and to gain confidence in the choice made.*

A bridge must be strong enough to bear its own weight and the weight of vehicles and persons using the bridge. A bridge must also cope with various weather conditions.

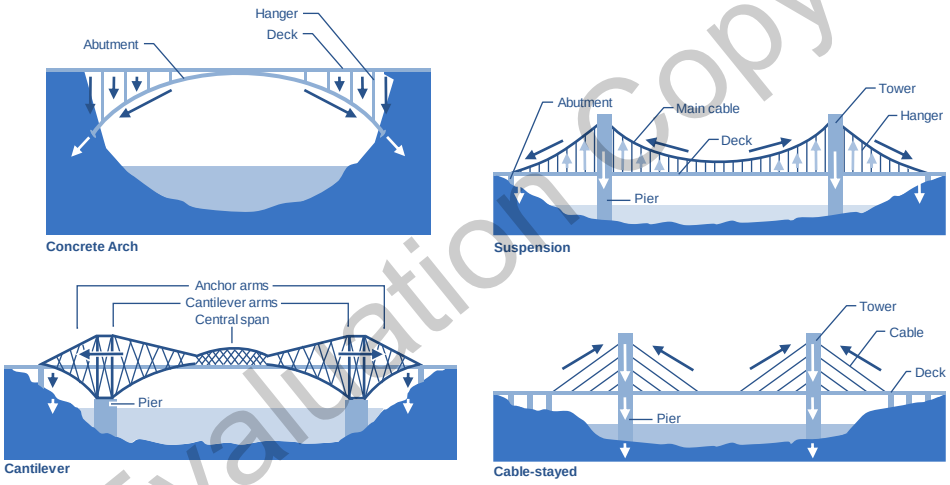

Figure 1.7. Schematic drawings of different suspension bridge concepts.

The designs of various types of bridges, which have other purposes, therefore have a different length. Figure 1.7 shows an example of four ways energy can be distributed. Anchors at the end of each bridge drain the power into the earth via (concrete) arched bridges or suspension bridges. Cantilever bridges have two independent cantilever arms that mirror each other and are united by a central span or via cables to the tower (Cable-stayed).

The schematic drawings of the different principles for various bridge concepts support clearly show how these concepts work. It significantly helps to understand and reuse the principle and qualitative advantages of the concepts in the bridge design.

1.5.3 Style elements define the architecture style

In order to be able to recognize concepts that underpin structures, structures must meet certain implicit requirements that are the result of the presence of concepts. Such as the presence of a number of issues that belong to a concept. These parts are called the elements of a concept. For instance, pillars, cables and towers are elements of the bridge concept. Cables and pylons are then elements of the cable bridge concept, which represent a specialization of the bridge concept. All elements within the concept have their own identity and functionality and certain behavioral aspects that result in qualitative advantages and applications.

Elements that often reappear in concepts are called style elements. On the drawing board, style elements can also be used to convey certain behavioral aspects and qualitative advantages of the concept in the design of a structure. Examples of style elements in physical structures are the pointed arches and rounded corners in a church with a Gothic architecture style.

A well known architecture style is rationalism. An important feature of structures built according to rationalism incorporate support structures that consist of brickwork.

For example, the personal architecture style of Frank Lloyd provides large areas in structures with open corners. In the corner of an area elaborate window constructions allow the outside environment to become part of the interior and the interior to become part of the outside environment. Concrete and steel make it possible for an overhanging construction is supported in a way that the corner remains 'open'.

Another architecture style is 'Form Follows Function'. Central to this architecture style is the idea that the design of a project is inspired by the utility or functionality of a structure. This architecture style uses different style elements from different architecture styles.

- *A style element is a frequently recurring element that is part of or belongs to a style. An element becomes a style element if it becomes part of or is seen as belonging to a certain style. This often is the result of the features of that element.*

1.5.4 Making a statement by the building architect

A building architect often expresses himself through his design. He has his own vision with regards to design, building or using structures or he has his own particular view of the world view or on life itself. This means he expresses those in his structure. He wants to tell something through the design of the structure.

Through structures, architects search for the operative, decorative and structural limits of concepts. When do we find something that makes us want to work in a particular building? Dare we sleep in a suspended house? Can we construct a bridge of paper strong enough to support traffic? With a paper bridge or a house of glass a building architect will make an architecture statement such as 'weak = strong' or 'I-have-nothing-to-hide'.

The building architect can make an architecture statement about someone else's structure. Suppose an architect appreciates the operational excellence of a conveyor belt in a food factory. As such he uses the concept of the factory and a conveyor belt as a blueprint, for example, to design a Japanese restaurant with a rotating table displaying a variety of food dishes. This way the restaurant guests are able to put together their bespoke meal on the basis of the standard food dishes on offer.

- *A architecture statement is an expression of a structural vision, philosophy, culture and identity that the architect makes through the design of the structure.*

1.5.5 Uniting conflicting requirements at conceptual level

A building architect wishes to create designs on the basis of sustainable and future-proof structures at the behalf of clients. The building architect is therefore constantly searching for new architecture design assignments with sufficient mandate to express his own vision about a structure. From his perspective, he is actually waiting for the right client to fulfill his dreams.

The design that the building architect creates is known as an architecture design, also called 'architectonic design'. The architecture design is a special kind of design. An architecture design typically focuses on a large complex consisting of many parts. The design often focuses on parts of a complex entity.

- *An architecture design is a schematic representation of a plan, or representation of a solution to*

a design issue. For instance, a structure that fulfill the needs of stakeholders. In an architecture design, complex or difficult design issues are solved with the help of concept and principles. Hereby functionality and form were designed separately as well as construction, decoration and operation of the solution. Altogether this makes a design, an architecture design.

For instance, to building architects architecture design issues could be:
1) How do I get an affordable design of a factory with today's technology, which is both environmentally friendly and animal friendly.

2) In order not to lose space and safety, how do I design an area that is separated from the outside world using very thin but strong materials; which, also gives me the possibility to look outside but not the other way around?

Architects make architecture designs, designers make construction designs. The architecture design is a highly visual design, with many different views and perspectives of the structure which can be visually examined from many different perspectives.

It is imperative that a building architect obtains his commissions for architecture designs, on the basis of the pre-condition that the design is suitable and usable; and that it appropriately visualizes the architecture design, to enable the client to make equitable choices.

- *An assignment for an architecture design is an assignment in which the architect is awarded a contract with sufficient mandate to make an architecture design of a structure or a part thereof. The client is often an owner of a structure in an existing situation or an owner of real estate of a new structure. The contract must also state how quickly a project must be completed and how much the implementation of the design may cost.*

The building architect always commences by making sketches of concepts, and abstract solutions, in order not to commit himself completely at the beginning and to retain the freedom to create a design. He often has to unify the conflicting requirements of stakeholders in his design. For instance: 'create an open but safe solution.'

A building architect must ensure that he does not receive an incomplete non-viable design assignment or that the client asks him to design something that can not be

implemented due, for example, to lack of ownership rights or financial means. An elaborately detailed design assignment is the basis for ensuring clarity.

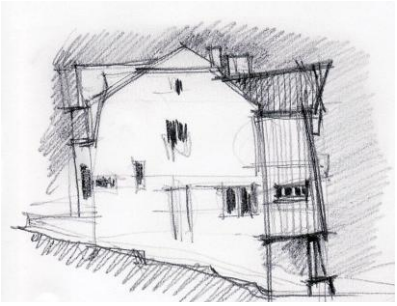

Figure 1.8. Sketch of a structure.

In order to design sustainable and future-proof structures and to be able to communicate about it, the building architect must adequately convey how concepts work and how certain advantages can be realized, i.e. to illustrate the design principles with sketches, drawings or cartoons of the principles.

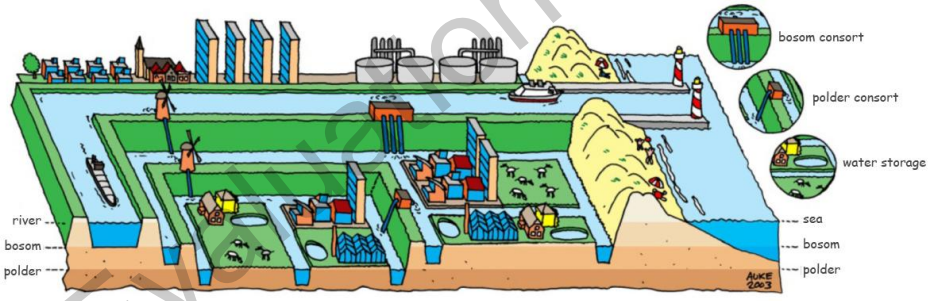

Figure 1.9. Principle cartoon of the polder canal system.

Also, the building architect draws large design drawings, called blueprints, which show contractors and builders how everything fits together so that they can, for instance put together a phased plan better for building a structure. In these drawings, the architects use different views of the structure.

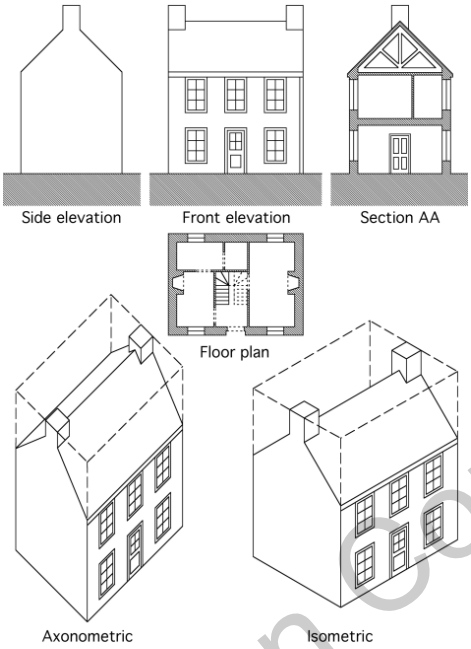

Figure 1.10. Standard views (views) that are used in design drawings and construction drawings.

At the bottom of Figure 1.10 are two examples of visualizations of a house where no vanishing point (perspective) is used, but where the sides are parallel. This kind of visualization is called isometric and axonometric projection. In axonometric visualizations three sides are always visible at an angle of 120 degrees. Such views are not used to show the whole, but some elements relative to each other in order to show where the parts themselves are visualized in full size.

- *A view is the interpretation of what a person sees or finds important by virtue of his function, concerns, knowledge and skills. A view is a subset of a model.*

For example, a gardener who looks after a garden will see more planting and maintenance issues than the owner of the garden. The difference is the knowledge that the gardener looks in particular, at certain entities of the garden in an expert way. They both look at the same garden model, but each pays attention to a different subset of entities. This subset of entities is called a view.

A building architect utilizes views in order to communicate with different target groups about different aspects of a structure. Hereby think, for instance, of a

functional view, a technical construction view, a user view and a management and maintenance view.

The building architect creates views such as: side view, construction view and structure vision. For an architect, a view is often the image of a structure from one side. If an architect places more than one side next to another then we call this a perspective. This perspective is further elaborated in the chapter 'Dragon1 Way of Representation'.

Figure 1.11. Exploded perspective of a house.

In Figure 1.11 an exploded perspective of a house is shown. This is a projection with depth, where it looks as if the parts of the structure are pulled apart. The drawing in Figure 1.11 gives the feeling of a construction drawing. Such perspectives are used to show a whole picture and how everything fits together, and how components relate to each other. Commonly used perspectives, next to the exploded perspective that we already mentioned here, are the cut-away perspectives, the cross section and the see-through perspective.

This way a building architect will create a requirement model, concept sketch, a design drawing and a structure vision in order to provide the client a good image of the end result at an early stage.

Later in the project, building architects often create blueprints for the people who need to implement the design and makes artist impressions for all possible

stakeholders to convey as far as possible the mood, emotion, feeling and appearance of the structure, for instance in lieu of creating commitment and support.

- *A blueprint is a detailed construction or assembly plan. A blueprint is a building or assembly view aimed at a structure on a two dimensional plane.*

Note: If a drawing of an aircraft consists of more than 10,000 parts, the pictures that are currently created by enterprise architects often consist of no more than ten items. The enterprise architect, however has much more control regarding the predictability and the behavior of a structure if he could introduce more parts into the design.

Building architects utilize a contextual interpretation to visualize the surroundings in which structures will be implemented that they design and to determine the principles and concepts present in a particular area. The context is then explicitly defined in the design in order to clarify what lays in and out of the context.

- *A context is a defined area around an entity that is taken into consideration so that the entity is given more meaning. In a context certain circumstances and events that are related to the entity are taken into consideration.*

Examples of a context: the center of a city where there is projected a new music theatre or to the government where a municipality hosting e-business services becomes the starting point of a chain of organizations.

1.6 A new conceptual framework for enterprises

It is now possible according to the acquired knowledge of building architecture and of building architecture concepts from the previous sections, to look at enterprises in a different way. This new vision, we call **Enterprise Architecture as Total Concept**, for Enterprise Structures.

1.6.1 The framework of Enterprise Architecture as Total Concept

Enterprise Architecture as Total Concept can in general described within a framework. The framework consists of eighty key terms in which the terms of building architecture are combined with these of enterprise architecture.

Those terms provide the architect a vocabulary to express certain issues and solutions in the architecture. 80 terms seem like a high number, but they form the basic vocabulary that every day, week after week needs to be used to do their job professionally.

In particular, the framework provides tools for concept designing and for making them visual and communicable. This makes Dragon1 unique and provides the architect the possibility to help clients with decisionmaking.

Dragon1 model – Enterprise Structures Total Concepts Framework (global)

Figure 1.12. Enterprise architecture as total concept framework (global).

Many of the architecture terms in this section are defined in terms that are very much related to each other. However in enterprise architecture, it is important to use the essential difference between these terms. As an enterprise architect one must have exact control over the designs and in the way that a business IT solution behaves once it is implemented.

1.6.2 A new view on enterprise architecture

Enterprises can therefore be regarded as systematic structures that consist of entities such as concepts, elements, components, objects and technical products. With this premise as a starting point, we look at the following example.

Looking at new way to enterprises, we will use 'BetterCare Ltd.', a care institution as an example. BetterCare Ltd. provides health services and contributes as a responsible enterprise in terms of risk, profitability and the survival of the enterprise.

- *An enterprise, such as a care institution or a bank-insurer, is a set of businesses that perform, in a certain way, a range for the market, by which they take consciously a responsible risk.*

The aim of BetterCare is to support elderly people to live longer on their own with a variety of tasks.

The enterprise architect carefully maps out the needs of the client. By making use of the phrase: 'I want ...' in order to formulate the client's needs. In Figure 1.13, a sketch is presented of a 'needs model' for the 'elderly' client profile.

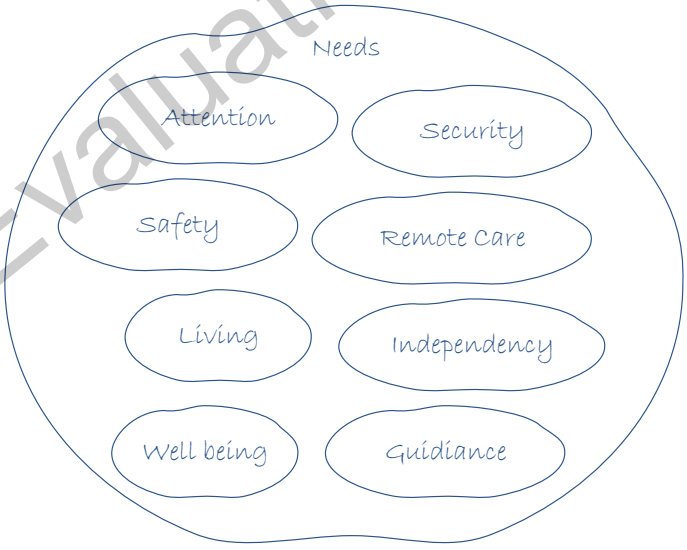

Figure 1.13. Sketch of a needs model.

Beside the needs, an accurate inventory of client requirement for care services was made. In general, an enterprise architect produces requirement models for several

stakeholders to get an overall impression of customer and employee wishes: 'how does one live and want to live, how does one work and how does one wish to work and how does one spend their leisure time and how would one wish to spend their leisure time?'

The enterprise architect, together with the Board of Directors and other stakeholders, bring together business and informatics concepts to see that the goals are implemented, the needs of the client are satisfied and the concepts fit the mission, vision and identity of the institution.

The first thing that the enterprise architect did is to bring the total concept in line with requirements of the client and the Board of Directors of BetterCare, who in turn approved it. For an outline of the total concept model see Figure 1.14.

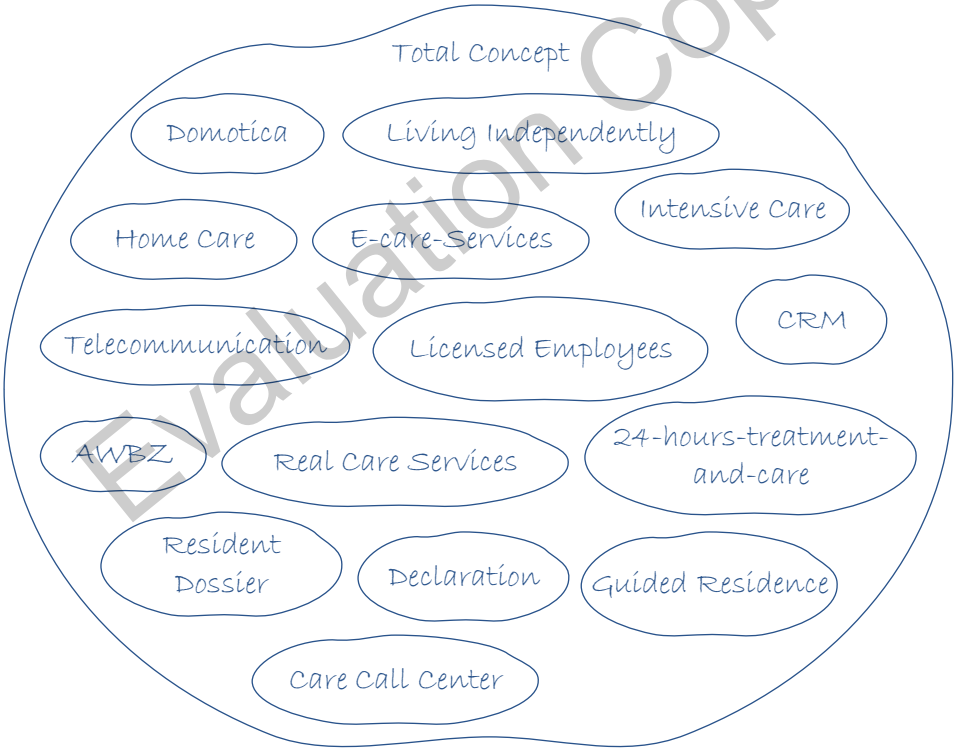

Figure 1.14. Sketch of the total concept model.

To achieve the main goals and fulfill the needs of the client, BetterCare has to offer certain home care services. These care services require much communication with the elderly living in their own homes. With BetterCare clients have chosen the home

health automation concept, consisting of in-home automation products ans services. The clients have chosen to link home health equipment with their own emergency room, which is operated by a 'specialized help desk' staffed with qualified employees only.

Added to this BetterCare will use of a new CRM information system containing a detailed dossier of those who live independently. The kept information is not limited to the residents but also includes contract details of their families.

Part of the enterprise architecture of BetterCare allows a conceptual description of physical and e-Care services, independent living, day-and-night care, and home automation, central help desk care with qualified staff, CRM and dossiers of the people involved.

Each of the BetterCare concepts can describe and visualize with the help of a concept principle. This will provide better understanding and communication. The architect could use the concept principle as a design principle in the design of the enterprise or at least in the specific section of the enterprise, which deals with home care services. The sketch of the concept that does not concentrate on the principle is called a concept sketch. A concept sketch emphasizing the principle, within the concept, is called a principle sketch.

Based on the residents' profile the architect has produced a design sketch, which includes the needs of the client, the total concept and mission, vision and identity of BetterCare, see Figure 1.15. The sketch constitutes elements and components, which are acknowledged by principles and concepts. In this design sketch there is no attention given for a specific situation of service or activity, but the different situations and activities have become more or less visible at the same time. A situation sketch could be produced for the specific situations mentioned.

All the concepts for BetterCare consist of generic elements and components such as certain types of products, services, processes, functions and applications. Interestingly, what shows up is that the chosen concepts consist of many standard elements and components in their own right. It is often the combination of concepts and configuration or collaboration of a concept, which in itself creates a unique total entity (total concept).

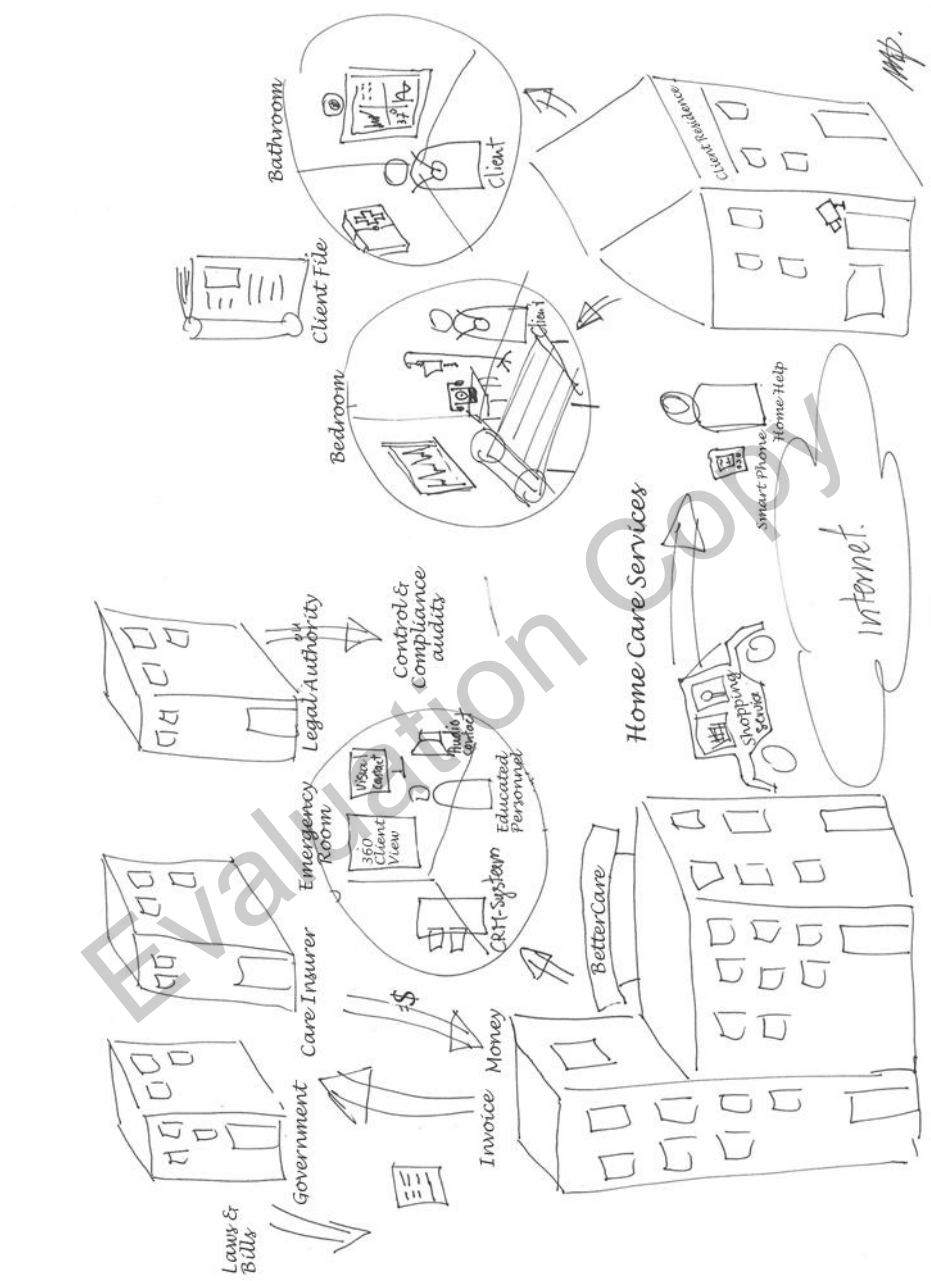

Figure 1.15. Example of a design sketch of a total concept.

Figure 1.15 shows the concepts, elements and components of the concepts visualized in real-life symbols and metaphors for these symbols. In Figure 1.15, three circles emphasizes on spaces and situations in the enterprise and the client home. The design sketch provides the client a clear direction. The client will get in a better position to submit a design assignment stating the requirements or to make a proposal for approval by the architect.

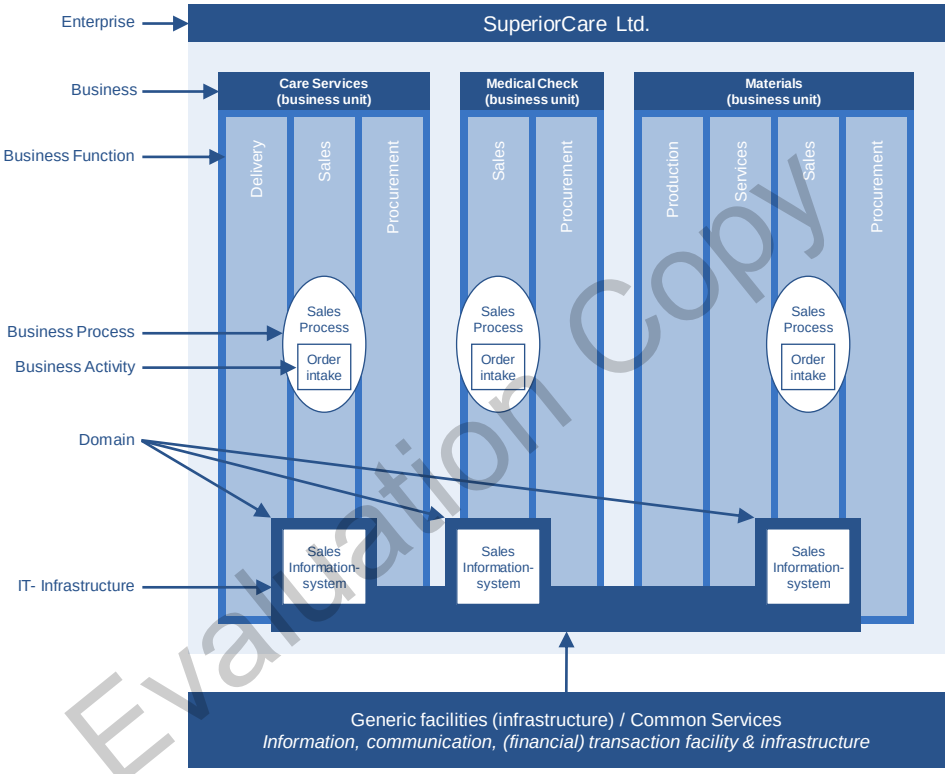

Figure 1.16. A structure vision with example configuration of BetterCare.

In Figure 1.16 the BetterCare care institution is divided into three business units to realize its goals: Care Services, Medical Check and Care Materials, whereby each business unit is organized into three or four business functions. Within these business units employees focus on common business objectives, which have been derived from the goals of the enterprise (which are binding above business level).

- *A business (unit) is an organization where people collaborate in business processes in which they create and deliver goods, products and/or services in order to collectively achieve results or to achieve goals concerning services and products.*

Examples of entities belonging within a Care Services business are: the business function, the care product, the online sales process and procurement system. Each entity is also regarded as a concept and form a part of the total concept and in turn part of the architecture. These entities are also to be considered as elements at a logical level in the architecture design of the enterprise.

- *An organization denotes a system to have certain facilities or features. Colloqually organization is often used as a synonym for enterprise. The best way to use this term is: 'The organization of ...' for instance a foundation, business, process, business, etc ...*

The business of BetterCare Ltd. constitutes of units in which work is performed. Work in a business must be carried out efficiently. This is also the case with BetterCare. Subsequently, a person works on the concept 'business function' and the concept 'domain'; enterprise architects at similar enterprises have had good results with this approach.

The Sales department of the business 'care services' focuses, among others, on the sale of different types of care services. The type and number of care services may vary. The fact that this department solely focusses on sales of care services is undisputed. This ensures efficiency.

- *A business function is a set of elements and components such as activities that are focused on achieving a common functional purpose. For instance, the business activity sales, with sales staff, sales processes and sales information systems.*

To keep work in a business function manageable, predictable and controllable by a person (the owner) activities are grouped and structured in business processes within business functions.

Within the sales function (business function) of a care services business, we recognize the business process as being the: sales process. All sales activities in this business function fall within this process. Besides the sales process, other related processes may be present in the business function, although they might not be shown in Figure 1.16. Keep in mind as an architect, that not everything should be brought into view (in detail), but only those which are important with regard to the design assignment.

There are different kinds of business processes, such as primary and secondary, that are directing, implementing and supporting business processes. All these kinds of processes exist within BetterCare, but are not visualized in Figure 1.16.

In an enterprise there are often many similar groups of business processes present in a business function. There are also business processes that transcend business functions. These groups form domains. However, the collection of certain types of resources for certain kinds of work can constitute a domain.

- *A business process is a structured time-sequential set of activities, with ownership, usually within the scope of a business function, but always across organizational units.*

Business processes ensure optimal coordination of activities across organizational units. As a counterpart to business processes, enterprises work also with work processes, for example on a department, to plan the work as optimally as possible. Each enterprise has the challenge of finding the optimal balance between excellent business processes and work processes.

- *A work process is a structured set of sequential time business activities, with ownership, usually within the scope of an organizational unit such as a department.*

Within the BetterCare care institution an example is given of how, sales activities take place within three business functions. These sales-related processes, activities and resources constitute the sales domain. The Chief Financial Officer of BetterCare is the appointed Sales domain owner. He is responsible for the coordination of business-wide sales activities. His ambition must be to have all sales activities to use a common IT infrastructure and information provision.

- *A domain is an (logical) area of responsibilities, tasks, roles, responsibilities and ownership.*

In an enterprise we identify domains because everything in an enterprise requires a domain owner; or else management simply would not be possible. Similarly within BetterCare we find the domain of 'Living at Home'. Everything relating to living at home is contained therein: the processes, information systems and data. Where more than one person claims ownership of an entity in the enterprise there will be an exponential resistance to change. Viewing an enterprise in terms of domains makes an enterprise more amenable to change. Within BetterCare it is clear. A

woman director is put in charge of the domain concerned with the customer details of people 'Living at Home'.

BetterCare provides care services to clients such as home-help for bedridden patients. This service aims at realizing a better quality of independent living at the behest of the client.

- *Service provision includes benefits, goods or products as a direct effect of activities or processes.*

The business responsible for the provision of care support products at BetterCare produces equipment such as crutches and custom made cleaning equipment such as a self-steering mini-vacuum cleaners.

- *Production is the process that results in a quantity of delivered products, goods or services.*

In an average enterprise there are thousands of business activities. The better these are visualized and given correct and understandable names, the better the architect can produce an architecture design to ultimately achieve optimal and efficient work activities. The sales process of the BetterCare services company has a business activity which is denoted 'order intake.'

- *A business activity is an activity (unit of work, tasks or actions) executed as part of a business.*

Clients are often the customers of BetterCare. However, sometimes the client's care insurers or administrators are the claiming and paying customers. A client is often the person or organization who may make requirements on the quality and performance of services and products, and who pays for the services and products.

- *A client is a paying consumer of goods, products or services, natural person or judicial entity.*

The customer process is a new way of looking at a series of activities. It represents a customer's external view. From a customer's process view, we discover where a client has to wait unnecessarily, or is kept in the dark about, the progress of affairs. At BetterCare they utilizes the customer process to assess how long the process of an application for the allocation of a house is taking when a client has heard nothing about the progress of the application for far too long. Now someone receives a progress call by telephone every week.

- *A customer process is a set of activities experienced, felt or seen within a certain period of time by a client. It is the customer's process view of the business.*

For users of a structure, an enterprise architect must primarily design provisions in spaces and at locations. Within BetterCare there is currently an IT infrastructure that exists entirely of fixed IT facilities such as an e-mail system and digital project workspace on the intranet. Everybody can now utilize the digital highway: this applies to staff as well as customers.

- *An infrastructure is a set of fixed services for common use, users have access to, but not have ownership.*

It is the enterprise architect's job to create an architecture design from all concepts and translate them into elements, components, objects and products to ensure that the goals of BetterCare are reached.

If the architect has little knowledge of the sector or the market in which an enterprise operates and has little insight into the business objects or information objects or the acceptable concepts and principles in that market, then it is almost impossible to produce a well formulated and correct architecture design. The enterprise architect at BetterCare combines three years experience as a nurse, fifteen years experience as an information analyst, four years as an IT manager and ten years as a project manager.

At BetterCare they started with a change project to work in a more customer-oriented manner. The project is executed under the principles of architecture and this is creating a fresh look at the CRM project at BetterCare. The usefulness and necessity for CRM and the changeover to source databases becomes immediately obvious.

An example of CRM for customer orientation at BetterCare

The strategic approach at BetterCare is to work more customer-oriented. The precondition is that the way this takes place has to fit the culture of the enterprise. The enterprise is going to make use of the concept of Customer Relationship Management. This concept revolves around just one element: 'a shared customer database is the only permitted source to contain customer details'.

The customer profile principle of CRM states that: 'By taking all data from a shared customer database, it will be ensured that staff attain maximum access to customer history

and attain a better understanding for the current customer situation whereby they are able to work in support of better customer-orientation'.

CRM is a concept that also exists outside BetterCare, however, with the use of specific components such as a Brand-Name database and a home-residential-business object BetterCare adds to this concept.

Enterprises work with management concepts such as ambitions and strategic starting points in order to provide frameworks to achieve targets. An ambition is a higher purpose where one assumes that this should be achieved. For instance: 'At BetterCare it is our ambition that we strive to provide the best products available to our customers.' An starting point is a base or fundament that should be respected when achieving ambitions. The starting point is for instance: 'By taking a lead in technology, we will become the best, are the best and remain the best, and through technological innovation in our products that our customers like to see.'

Enterprise architects want their clients to express themselves in additional terms regarding ambitions, starting points and preconditions concerning the strategy of the future design and implementation of the CRM-structure.

A manager of BetterCare wants to purchase the new CRM system. The purchasing department provides the prerequisite that the manager may only purchase this from preferred suppliers; otherwise the investment is not covered by insurance. The IT department makes it known that only standard systems should be purchased, otherwise they will not manage and maintain the system.

- *A precondition is an additional requirement or condition which is necessary in order to achieve a goal. It is an external requirement for most other stakeholders. A minimum requirement set by third parties, whereby a minimal compliance is to be expected.*

Using the BeterCare example, we try to explain how an enterprise architect benefits from enterprise architecture, just as the building architect benefits from visualizations in building architecture design. Using visualizations of the current organization of the environment and using visualizations of a solution consisting of concepts and principles, the building architect is able to communicate towards the client to what extent his strategy can be realized. This allows the client to gain insight and overview, enable him to take the decision for the realization of the solution and with that for a fundamental change in the environment.

1.7 Questions

After studying this chapter, the reader is able to answer the following questions.

1.1. Describe how we perceive architecture in general.

1.2. Describe the role of the architect. What is an architect and what are his main tasks?

1.3. Name some building architecture terms that are taken into enterprise architecture as total concept for enterprise structures.

1.4. Explain what a concept is.

1.5. Explain what a style element is.

1.6. Explain what a principle is.

1.7. Explain what a concept sketch and principle drawing are used for.

1.8. Explain what constitutes a structure and why an enterprise is viewed as a structure in Dragon1.

1.9. Name the single model and the basic 10 enterprise diagrams and solution diagrams Dragon1 recommends to every architect to alway create in every assignment and for every project.

1.10. Name the two toplevel products an architect always creates and the five generic architecture visualizations that are always present in these products.

Chapter 2

Dragon1 at a glance

Always keep in mind that a method provides a common glossary of terms and an approach to work more effectively and efficiently; a method alone should never be a goal.

2.1 Study Objectives

After studying this Chapter, the reader will be able to:

- state the benefits of Dragon1 as an open method.
- state the meaning of working with architecture.
- state the essence of the Dragon1 Way of Thinking.
- state the essence of the Dragon1 Way of Working.
- state the essence of the Dragon 1 Way of Representing.
- state the essence of the Dragon1 Way of Supporting.

2.2 Introduction

This chapter, 'Dragon1 at a glance' provides an overview of the entire method Dragon1. In this chapter, the most important Dragon1 concepts, and subconcepts are introduced as well as corresponding models and definitions. The chapter consists of schematics and broad outlines to be read as a summary of the entire Dragon1 method.

In the following paragraphs Dragon 1 outlines how management and architects can use Dragon1 to organize working with architecture, to design, control, adapt as well as business transformations and innovations within an enterprise.

The Dragon1 concepts outline how the method is constructed. The four 'ways' represented in Dragon1 are: the way of 'thinking', 'working', 'representing' and 'supporting,' are covered systematically.

2.2.1 What is a method?

A method enables people to achieve controllable and transparent results in a structured and repeatable way. A method is an efficient way to manage and control the quality of the results in which something is executed. Everything that is executed and produced is defined and standardized, which as such requires a method with a focus on three aspects: the function of method, useful to a large number of people; the form of a method to make it accessible for many people; and the structure of a method, to make it teachable, communicable and implementable by many people.

- *A method is a structured approach, used by an individual or a group enabling them to cooperate more effectively and efficiently to achieve a common result.*

Dragon1 model – Method Meta model

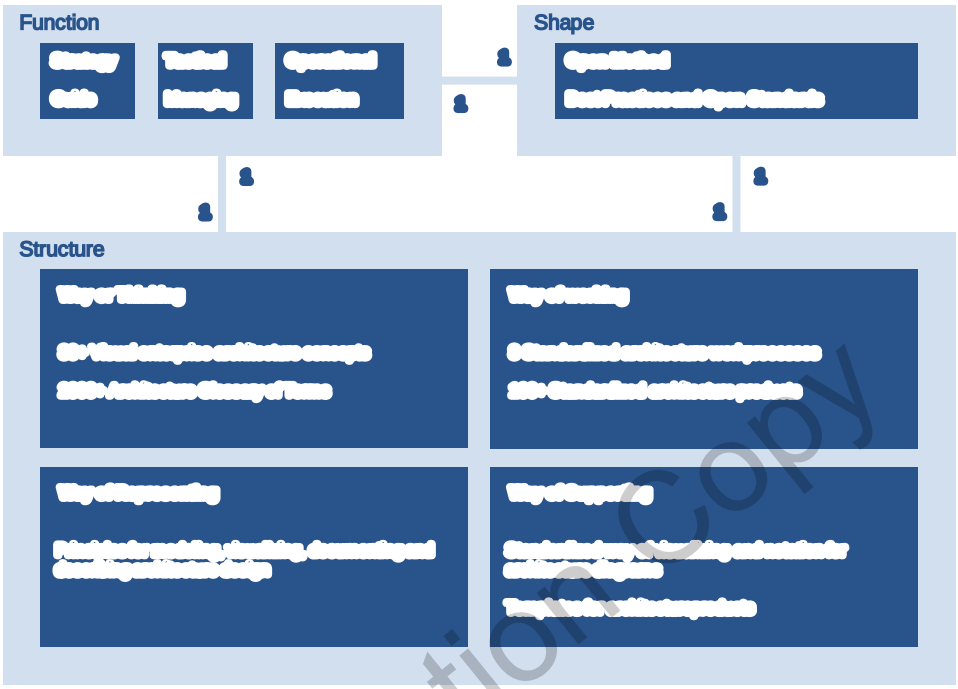

Figure 2.1. Method Meta model.

The **Method meta model** shows by means of an example how Dragon1, at strategic level, provides direction on how to work with visual enterprise architecture. Also, how at tactical level, Dragon 1 contributes to the way visual enterprise architecture is managed. At operational level, Dragon1 contributes to the implementation of architecture role activities such as those shown in the architecture work processes.

Dragon1 is an open method. In fact, the method is a cluster of over 20 Dragon1 open standard documents, each one supported by best practice documents.

The content of Dragon1 consists of a methodical structure that is divided into four 'ways':

1. The 'way of thinking' of Dragon1 describes the way that architects and stakeholders look at architecture, design and enterprise and what they observe as a result.

2. The 'way of working' with Dragon1 describes the practical application of working with visual enterprise architecture, and the way it can be adopted and adapted by large and small enterprises

3. The 'way of representing' of Dragon1 describes the way to conduct principal-based visualization of models, views and perspectives as directed and adapted to communicating with target groups.

4. The 'way of supporting' in Dragon1 covers practical tools, such as documentation templates and notation formats to enable the architect to generate better quality architecture products in less time.

2.3 Working with architecture

Working with architecture the Dragon1 way entails structural, systematic and planning of working with visual enterprise architecture. For instance, the realization of architecture designs for a strategic change management program. As such, architecture products are created in a controlled fashion.

Working with architecture, using Dragon1, a management process system can be vertically implemented at a strategic, tactical and operational level in an enterprise.

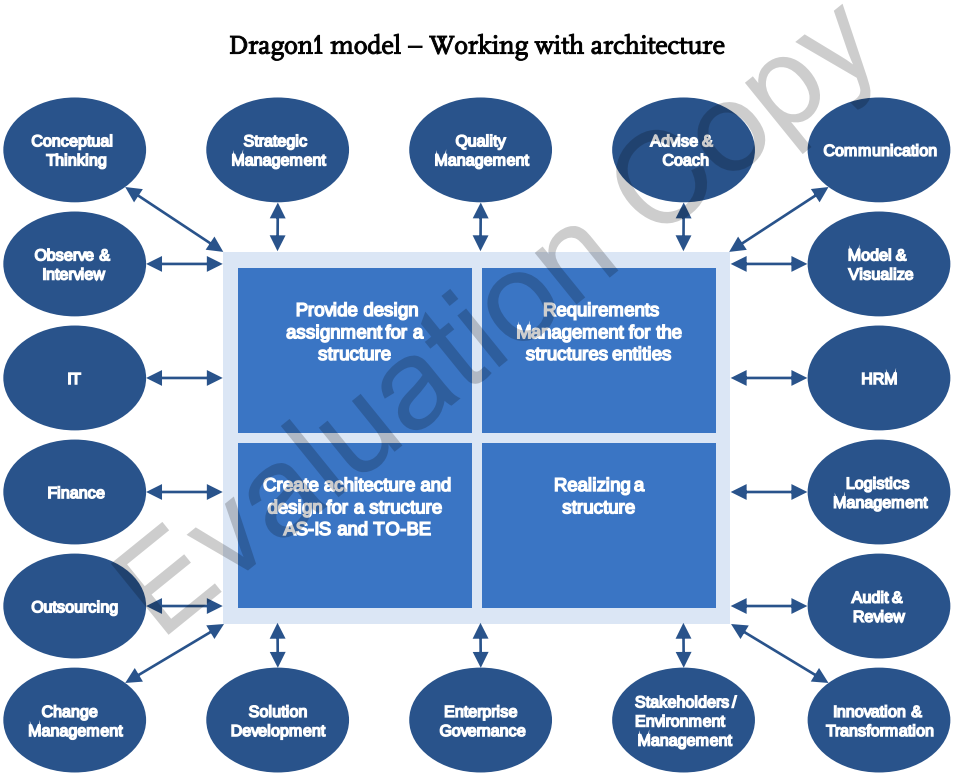

Figure 2.2. Working with architecture.

In the W**orking with architecture model** an example is given how business and informatic functions, processes and activities in the enterprise are linked together in an integrated manner with activities that make it possible to work with architecture.

Working with architecture means working with a discipline, which aims at finding solutions for strategic and change management issues with regards to enterprises from an architectural way of thinking.

- *Working with architecture is an integrated process of realizing business transformations and IT innovations in enterprises. Hereby the provision of design mandates, the determination of requirements, the creation of architecture and design as well as the realization of structures must be minimally present.*

  Working with architecture consists of a total of work processes and activities, which result in good, usable and adaptable architecture designs. Working with architecture ensures adaptability and visualization of the current AS-IS enterprise architecture and the realization and application of a TO-BE enterprise architecture.

Working with enterprise architecture allows management to do two things simultaneously:

1. To provide randomly and at any given moment, visible and clear architecture to an enterprise or part thereof. On the basis of this, the relation to the strategic assumptions, business aims and requirements become clear. And in turn, the quality, pertaining to usefulness, adaptivity, beauty and robustness as well as quantity pertaining to complexity and scale of the enterprise or parts thereof become clear.

In Dragon1 we consider business architecture, information architecture and technical/IT architecture as part of enterprise architecture. These architectures can also be made randomly, and at any given moment become visible and clear, but preferably only as a result of an architecture design assignment.

The CXOs, directors and managers are able to make decisions on the basis of qualitative and quantitative perspectives emanating from architecture. This is to the advantage of change management initiatives, strategy, policy, business operations, projects and programs. Management activities take place on the basis of these perspectives and the overview of the architectures.

2. To design architecture in support of a plan, idea, change, program or project, which becomes an integral design of a structure and as such it becomes a realizable, sustainable integral solution.

By producing a design for an integral solution on the basis of the scope of enterprise architecture, the environment of the solution becomes an appropriate part of the design. Hereby a solution can be better implemented and integrated in its immediate environment. The architecture design is as such produced and used for the purpose of high quality solutions.

Dragon1 typifies the activity of working with architecture such as producing architecture designs and using these designs for the purpose of showing the impact of enterprise architecture and using this insight for the purpose of decision making.

According to Dragon1, the design of an appealing innovative technical solution to support challenging functional issues, is a characteristic of working with architecture. This typifies the difference between normal design work and working with architecture.

2.3.1 Dragon1 Fundaments

Architecture can be used for a multitude of purposes, but for Dragon1 the emphasis is on visualization of the enterprise architecture, i.e. to use architecture for the design, control and change of an enterprise.

Dragon1 model – Dragon1 Fundaments

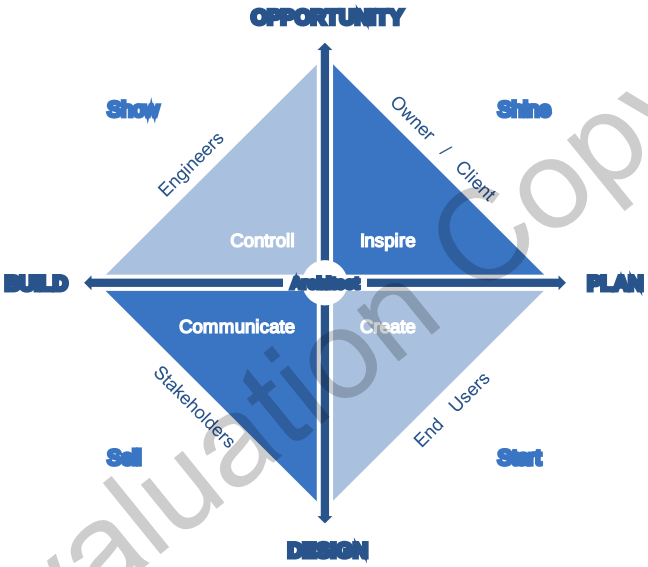

Figure 2.3. Dragon1 Fundaments.

The **Dragon1 Fundaments model** is linked to processes, services and organizational roles. It is an example of how architecture stands central in devising plans, the creation of designs as well as the realization of these designs.

In every direction architecture provides guidance to be used by groups of stakeholders – this is shown in the model along the angled shape Figure 2.3. For instance, architecture enables a client to devise a master plan, and make solving various issues possible.

Dragon1 model – Enterprise system architecture

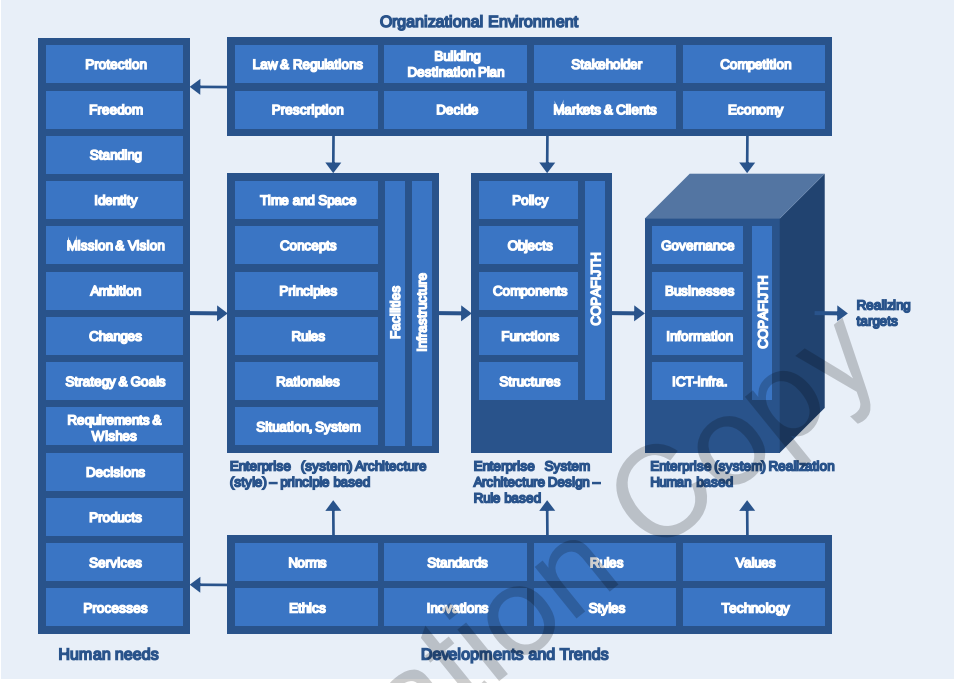

Figure 2.4. Enterprise system architecture.

The **Dragon1 Enterprise System Architecture model** shows how enterprise architecture is seen as a management instrument to how an enterprise is configured and arranged as a system. Also shown, is how architects include human needs, global development and environment of the business when making architecture designs and architectures. The theory of the holistic approach and the contingency theory constitute the basis to view enterprises in this way.

2.3.2 Design

In Dragon1 architecture is seen as a strategic management instrument to arrive at producing appealing and understandable architecture designs.

From a design point of view, Dragon1 considers it unique and essential, that no specific concepts, elements, components or technical products are prescribed to produce an architecture design. An architect must based on the client's

requirements and his own experience, knowledge and ability as well as the current level of technology and science, design a total concept and more elaborate architecture designs.

Dragon 1 provides thereby a SPACE architecture style, in which to find extant entities such as business processes, services, actors, applications, databases and life-events. However, the architect can also, on the basis of Dragon1 produce entirely new and innovative architectures, e.g. an enterprise without business processes, without computers and without money. This makes Dragon1 universal and very future proof. In Dragon1's meta model the enterprise entities are not included, but they are, however included in the SPACE architecture style meta model.

The fact that an architect designs based on experience places extensive demands on his competence. The architect's design must exceed the cut-and-paste level. Consequently a lot of attention in Dragon1 is given to architecture design and what designing actually means, but also how step-by-step an architecture design as a part of a total concept is produced. Visual enterprise architecture provides increased future enterprise do-ability. Consequently it must fit the planning and control cycle seamlessly as well as innovation within the enterprise and the way the enterprise is managed.

Dragon1 model – Enterprise Innovation Method

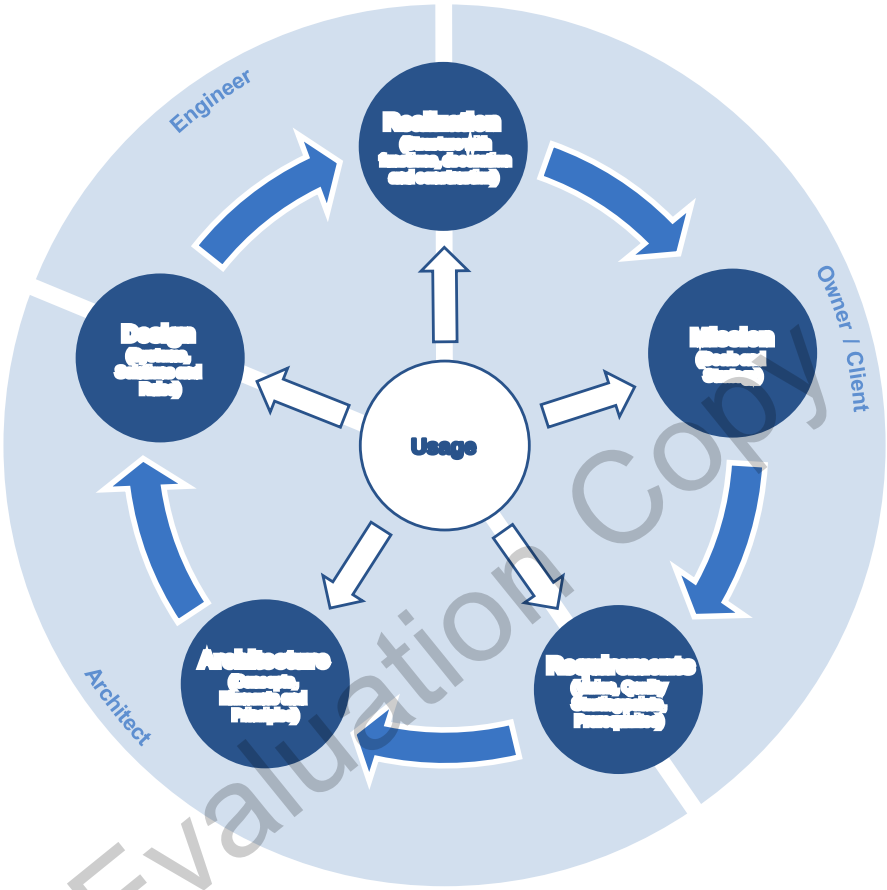

Figure 2.5. Enterprise Innovation Method.

The **Dragon1 Enterprise Innovation Method model** is an example how architecture has a positive influence between setting requirements and creating architecture design, to accomplish high quality solutions.

The client's enterprise mission statement provides a framework for his requirements – these requirements are translated by the architect into architecture and design, on which basis, a new enterprise or parts thereof are realized. The client is involved in using the realized solutions, mission statement, vision, strategy and goals, upon which the requirements are based.

The architect is involved in the realization of solutions by means of architecture design, which allows the architect to include requirements. The architect enhances to requirements by selecting concepts, stylistic elements and principles. As such, it becomes clear that two architects although they are given the same requirements, will create a different architecture with different concepts and principles.

2.3.3 Control

In Dragon1 architecture is also seen as a strategic management instrument to manage programs and organizations.

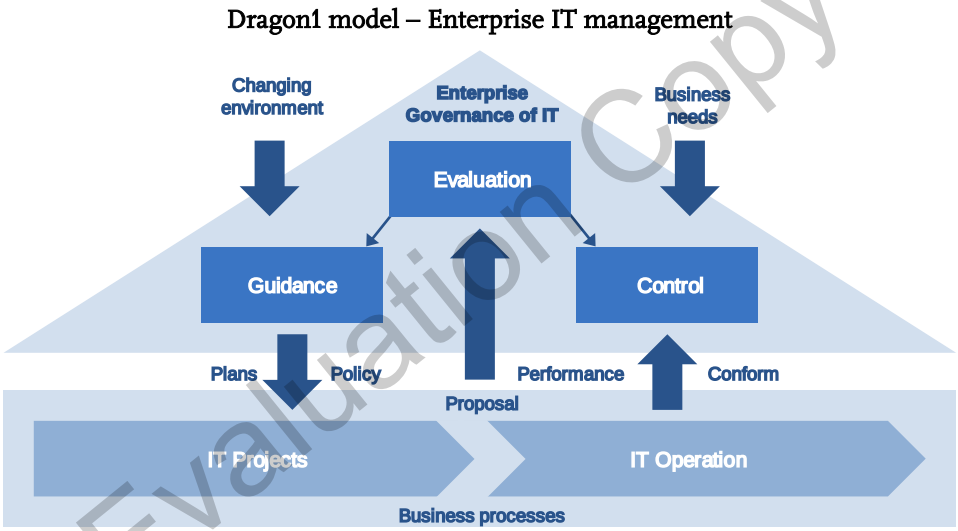

Figure 2.6. Enterprise IT management.

The **Dragon1 Enterprise IT management model** provides an example of how, with the support of business plans and change management plans, IT projects are provided by frameworks to ensure that these projects are implemented conforming to a business policy.

Architecture shows which standards, norms and directives must be used. An architecture document is hereby in principle a design codicil in support of a business policy.

Dragon 1 model – Enterprise Management Reference model

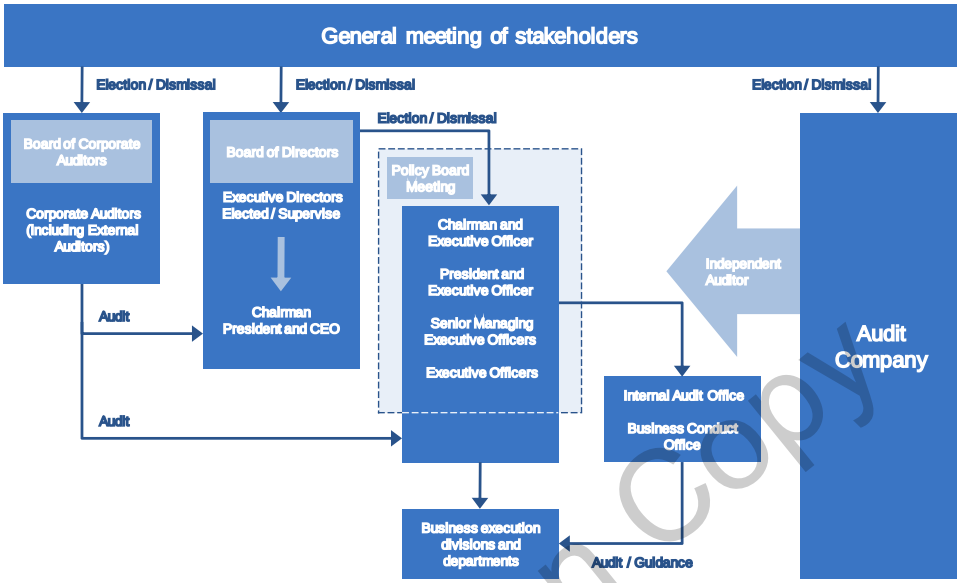

Figure 2.7. Enterprise Management Reference model.

The **Dragon1 Enterprise Management Reference model** provides an example of how managing an enterprise is subject to the relations that certain parties have with each other. Architecture fulfills a role in the management of enterprise, such as when auditors test to what extent components of the enterprise are realized in conformance with the required architecture parts as laid out in the reference architecture model. Subsequently management proposals are formulated on the basis of architecture. These proposals are broadly used by directors and management to adjust work and developments.

2.3.4 Change management and Innovation

Last, but not least, architecture according to Dragon1 is a strategic instrument to support changes (modifications), innovations (fundamental restructuring), transformations (identity changes), and improvements (optimizations).

The **Innovation Process model** provides an example how during the enterprise innovation process, architecture is used as input by means of concepts and principles to support the generation of new ideas and new products, as well as

process and services design. When thinking about and implementing innovation, in fact, managing innovation, architecture frameworks can be used as binding.

Architecture has a supporting role enabling the structuring of innovation. The Innovation Process model shows which type of information the innovators must incorporate.

Dragon1 model – Innovation Process

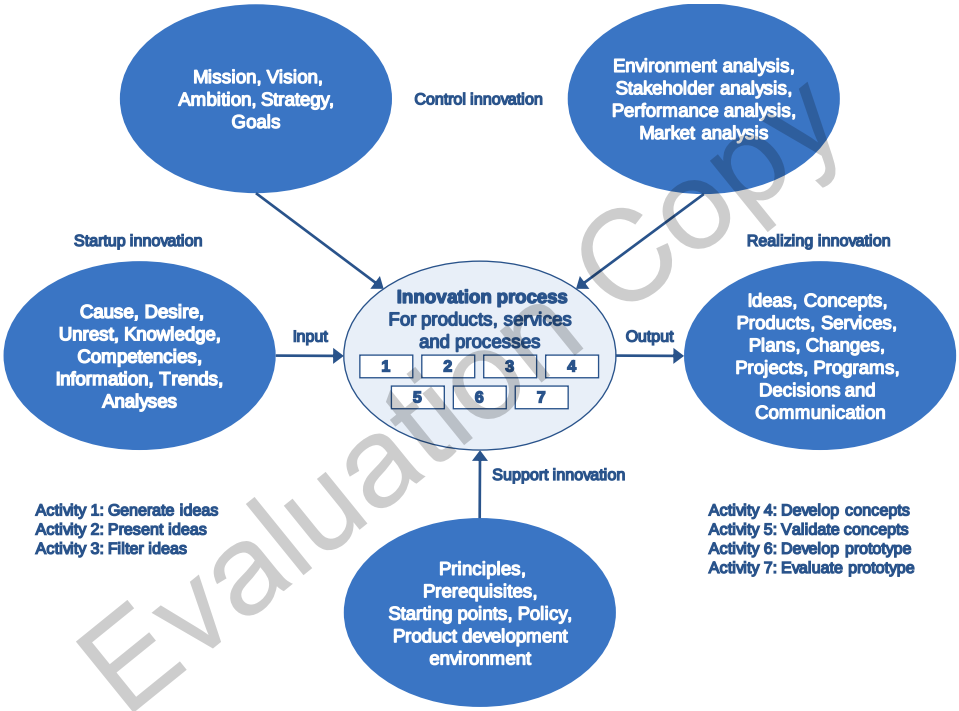

Figure 2.8. Innovation Process.

The **Innovation Process model** shows that the innovation process consists out of seven activities whereby the seven activities are iteratively implemented by the architect. In the innovation process environment, there are five other important areas. Certain matters provide the 'why' and the 'what' needs to be innovated. These matters are part of the environment of the innovation start-up, whereby the innovation process has different forms and types of end result. This is reflected within the confines of realizing innovation as an output of the innovation process.

In the top part of the model, it is indicated that the environment and the strategic level of the enterprise controls the innovation process by limiting space and providing scope and focus.

2.4 Method

The structure of the Dragon1 method includes four 'ways' that are briefly described in the model **Four ways.** Dragon1 recognizes the way of thinking: 'How do we view the enterprise and enterprise architecture themes?' What is our vision regarding these themes? Then, there is the way of working: 'How and in which way do we implement architecture process activities and which results or architecture products will this produce?' The way of representating: 'What, exactly, do the results of visual architecture look like?' How are they constructed, and what is their way of support: 'Which types of architecture tools are available to the user of the method?'

- *A method's 'way' the part of a method that focuses on a certain aspect.*

Dragon1 recognizes four ways, because as such, a methodical structural way can be described. For instance, if the architect has to accomplish something, like executing an activity or realizing a product, he will ponder about it in a specific manner. He will work at it in a specific manner, he will represent the result and the final process in a specific manner and he uses supporting tools in a specific manner. All these aspects accommodated individually in a number of Dragon1 concepts.

Dragon1 model – Four Ways

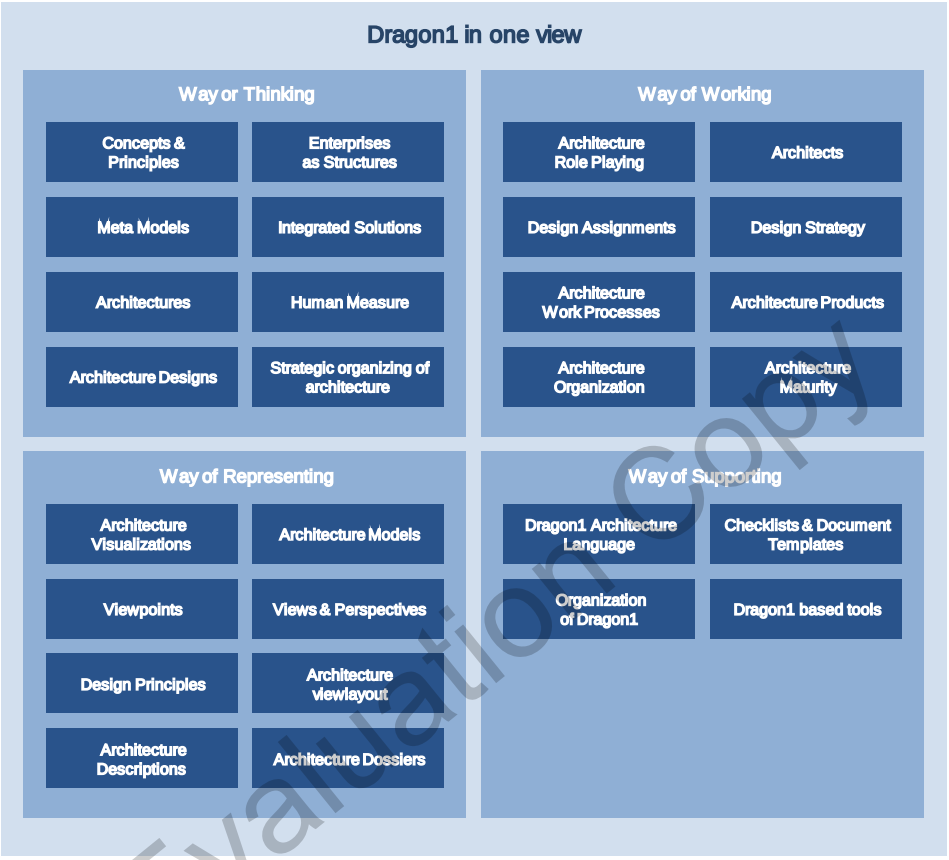

Figure 2.9. Four Ways.

We recognize Dragon1 concepts by their ways, because they are business and information scientific concepts, which have been made part of Dragon1 in the context of visual enterprise architecture. Regarding their ability to control and because of their transferability, these ways are divided into four to eight Dragon1 concepts. The Dragon1 concepts can, considering size, granularity and scope, be reapplied to aspects in business and information science.

The Dragon1 concepts themselves consist, in turn of part concepts. In this chapter, an overview is given of the method-ways within detail frameworks and within part concepts. In the following chapters of this book, the part concepts are elaborated upon in different paragraphs.

- *A Dragon1 concept is a concept that is a part of a Dragon1 way. A Dragon1-concept is regarded an important theme or subject in a method way.*

Other ways that are part of the Dragon1 method are: the controlling way, the managing way, the implementing way and the modeling way.

The way of controlling the method

The way of controlling is about the way in which Dragon1 is made available to users and the way by which users have influence on the ongoing development of Dragon1. The way of controlling the method, the way in which ongoing development is structured in the method and transparently designed, is found in the way of supporting.

The way of managing the method

The way of managing in Dragon1 is about the way an enterprise can efficiently and effectively organize working with architecture as well as manage architects. The way of managing the method is, as the management and control process, is found in the the way of working.

The way of implementing the method

The way of implementing in Dragon1 is about the way an enterprise can efficiently and effectively carry out architecture work in Dragon1. The way of implementating the method, as the implementation process, is found in the way of working.

The way of modeling in the method

The way of modeling is about the way Dragon1 is viewed and the way modeling is prescribed and which sorts of models are recognized. The way of modeling the method is for a large part found in the the way of representating.

2.4.1 Dragon1 Way of Thinking

The first part of Dragon1 is the way of thinking. The way of thinking of Dragon1 consists of eight Dragon1 concepts. Dragon1 has a unique and distinct vision regarding each concept.

- *A way of thinking is an aspect of a method that includes different visions and views regarding various themes or concepts.*

Dragon1 model – Way of thinking framework

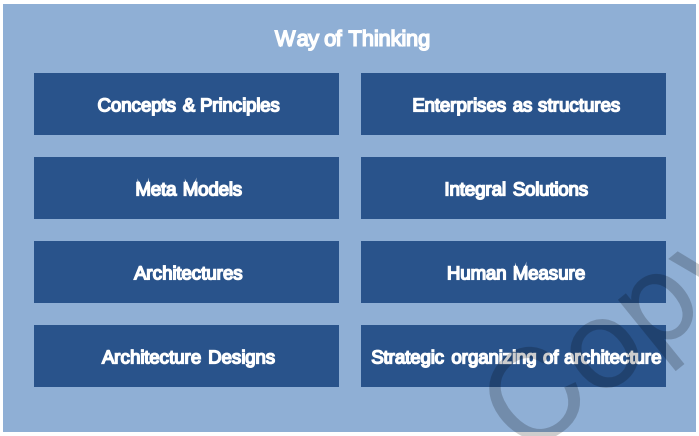

Figure 2.10. Way of thinking framework.

The model **Way of thinking framework** shows how eight concepts collectively constituting the way of thinking. In accordance with Dragon1, the architect must, when creating architecture designs and determining architectures view concepts, principles, meta models, architectures, designs, enterprises, people as well as working with architecture in a distinct way. The way in which Dragon1 views this aspect is described in Chapter 'Dragon1 Way of Thinking'.

Dragon1 – The way of thinking detailed framework

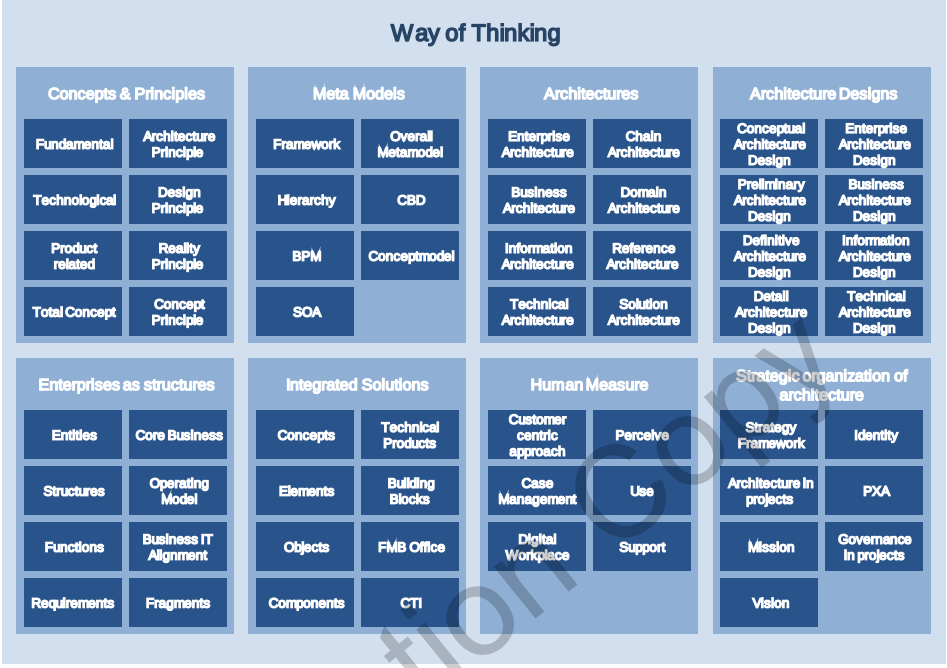

Figure 2.11. Way of thinking detailed framework.

The model **Way of thinking detailed framework** contains part concepts of the Dragon1 way of thinking. These part concepts are elaborated upon in as many separate paragraphs as possible in the chapter 'Dragon1 Way of Thinking'.

2.4.2 Dragon1 Way of Working

The second part of Dragon1 is the way of working. The way of working of Dragon1 consists, like the way of thinking, of eight Dragon1 concepts. In each concept, Dragon1 shows by way of examples how people use architecture in an enterprise. The way of working is covered here in an abridged manner.

The model **Way of working framework** shows which eight concepts collaborate to form one way of working method. When creating architecture designs and determining architectures, the architect must fulfill his role in the architecture role-play in accordance with Dragon1. In the same way, the architect must go after design commissions, work in accordance with standardized work processes as well as

produce architecture products from different architecture services. As such, an enterprise is enabled to properly prepare, organize and facilitate itself.

- *A way of working is an aspect of a method that covers different ways of work and approach regarding various themes and concepts.*

Dragon1 model – Way of working framework

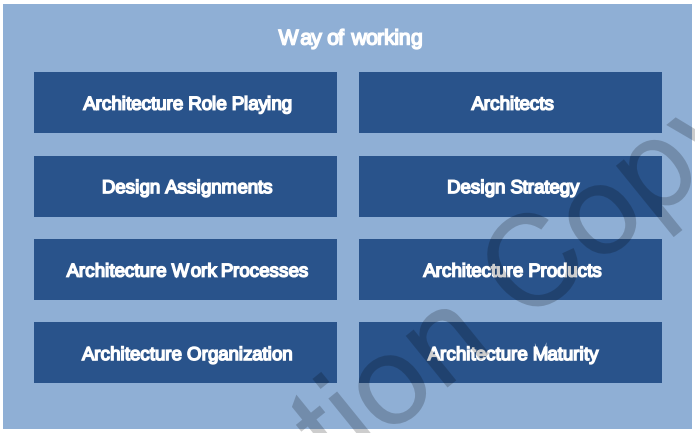

Figure 2.12. Ways of working framework model.

For instance the concept 'Architects' emphasizes the architecture role: the architect. The architect is the one who is involved in important parts of the architecture work, such as creating architecture designs, leading to different kinds of architectures. Dragon1 provides an idea how an architect must view architectural work. What does this mean to the architect in terms of organizing his activities? And what does that mean in terms of architectural services he can offer an enterprise?

The concept 'Architecture products' is also seen as a separate concept, because this concept entails a number of important products, which emanate from architecture work processes. The architecture products that are mostly created by architects and which carry high quality requirements are described as an architect's disciplinary products.

Types of architects

Dragon1 differentiates between various types of architects. In a practical sense, other types of architects are recognized also, but the architects that are described here are known for their work within enterprises.

- Enterprise Architect (EA). Creates architecture designs of (or parts of) enterprises, chains of businesses, businesses, detailed business functions and supply of information as well as IT infrastructure. EA activities revolve around reference architecture and enterprise architecture. The enterprise architect directs the realization of enterprise architecture designs in programs and projects, and on behalf of the client he provides governance on the end result.

- Business Architect (BA). Creates architecture designs of (or parts of) businesses, business functions and business processes and the supply of information, information systems as well as IT infrastructure and networks. BA activities revolve around reference architecture and business architecture. The business architect directs the realization of the business architecture designs in programs and projects and on behalf of the client he provides governance on the end result.

- Information Architect (IA). Creates architecture designs of (or parts of) information facility, information systems as well as IT infrastructure, platforms and networks. IA work revolves around reference architecture and information architecture. The information architect directs the realization of information architecture designs in programs and projects and on behalf of the client he provides governance on the end result.

- Technical Architect or IT Architect (TA). Creates architecture designs of (or parts of) the IT infrastructure, platforms and networks and the information supply and information systems. TA work revolves around reference architecture and technical architecture. The technical architect directs the realization of technical architecture designs in programs and projects and on behalf of the client he provides governance on the end result.

Likewise, Dragon1 recognizes the management architect, the application architect, the security architect, the chain architect, the human capital architect and the financial architect.

All architects, irrespective of the type of architect they are, aim at innovation, a reduction in complexity, a decrease in costs, etc. The resources used by almost every architect revolves around coherence, consistency and integration of different kinds of solutions into sustainable, future-proof, resistant, functional and beautiful entities.

Regardless of the type of architect, there are generic architecture products that are produced by these architects. Later in this chapter we will further broach the subject of architecture products.

A practical ten-step plan

To create an architecture design, Dragon1 has developed a practical ten-step plan to enable an architect to work with focus.

1. Ensure that client and design assignment contain purpose and a mandate. Test the quality of the final result by way of examples. Regarding the client, the architect must verify the client's requirements by enquiring which results he is envisaging. As an architect, to provide examples of possible results to inspire the client if the client is finding it difficult to put requirements into words.

2. Take care that as an architect you are well aware of the image of the stakeholders and that you have sufficient support from these stakeholders. Communicate with them regarding your chosen approach.

3. Ensure there is a program of requirements, compile these requirements by conducting workshops and use hereby, small visualizations to show the impact of requirements on a current environment.

4. Define and describe all concepts, elements, components, objects and technical products, which are subject to requirements.

5. Make a choice of the concepts and principles that could be used as solutions in support of architecture design. Create a meta model showing the relationships between chosen concepts.

6. Determine which rules provide support to design principles. Create a design on the basis of rules and design document criteria as appropriate.

7. Produce different drawings of the total concept for different types of stakeholders, in which it is shown how the requirements are addressed.

8. Define, together with stakeholders, the four phases of the total concept as a realizable unit. Use the principle 'by going back one step, one can later take two steps forward.'

9. Optional: Describe specifications and conditions by which the total structure must be built. Provide reference to standards, techniques and technologies. The architect circumvents with this step many risks regarding a contractor's incorrect interpretation of an architecture design.

Architecture products

In this paragraph, we look at the products which are produced by the architect. A number of these products consist of parts of products and these could also in turn consist of product components.

An architect should always be clear about when, as well as which products, and their corresponding quality he is producing, and for whom, in turn what the recipient is planning to do with it. In chapter 'Dragon1 the way of representating,' decomposition of a few of these products is elaborately discussed.

- *An architecture product is a professional, management or development product produced by architects or others, which play an important role in the creation of an architecture design.*

In the architecture processes of Dragon1 products are produced. In the model **Architecture products** a possible format is shown with regards to the most important products in an architecture process.

Figure 2.13. Architecture Products.

Project oriented working with architecture

In reality we often see that architecture projects start at an operational/physical level. That is the lowest maturity level recognized by Dragon1. The board and management do not possess visibility or exercise control over these architectural activities. There is a danger that strategic choices of architecture design are made by external bodies

lacking a total strategic vision of the enterprise. This often this leads to sub-optimalization at enterprise IT level.

2.4.3 Dragon1 Way of Representing

The third part of Dragon1 contains the way of representing. The way of representing consists, like the way of thinking and the way of working of eight Dragon1 concepts. The way of representating in Dragon1 clarifies the way an architect represents solutions in architecture designs to enable the client to make appropriate decisions.

- *The way of representating covers an aspect of a method involving different ways of representing, e.g. visualization on the basis of different concepts and principles.*

Dragon1 model – Way of representing framework

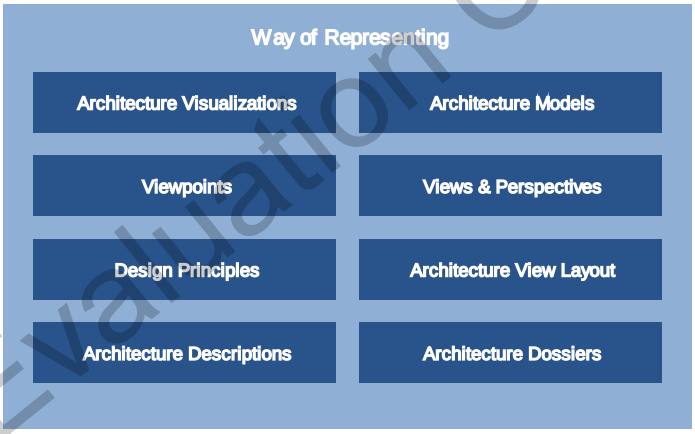

Figure 2.14. Way of representing framework.

The model **Way of representing framework** shows how eight concepts collectively constituting the way of representating. When creating architecture designs and determining architectures, the architect in accordance to Dragon1 produces certain visualizations in a specific way; he create models, including points of view, create perspectives, using design principles and he produces descriptions. The way in which Dragon1 advises how this should be done is described in chapter 'Dragon1 Way of Representing'.

Dragon1 model – Way of representing detail framework

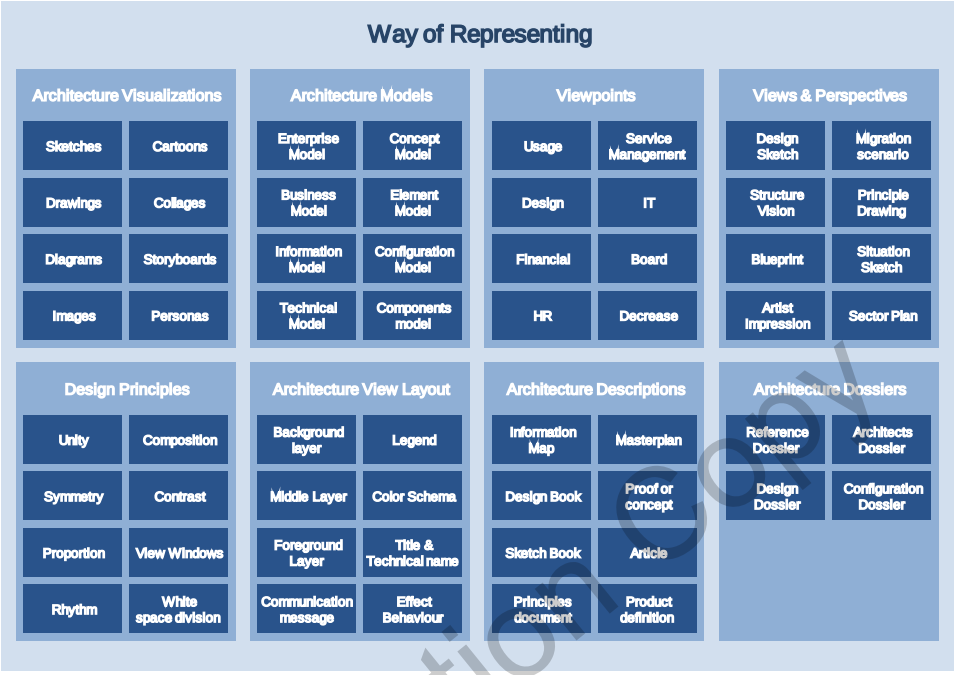

Figure 2.15. Way of representing detail framework.

2.4.4 Dragon1 Way of Supporting

The fourth part of Dragon1 describes the way of supporting. The way of supporting of Dragon1 consists, like the way of thinking, the way of working and the way of representing, of a number of Dragon1 concepts. In this instance, it consists of four Dragon1 concepts. The way of supporting makes clear how Dragon1, supports working with architecture using a number of support tools.

- *The way of support is an aspect of a method that covers the different ways of support with different types of tools and concepts.*

Dragon1 model – Way of support framework

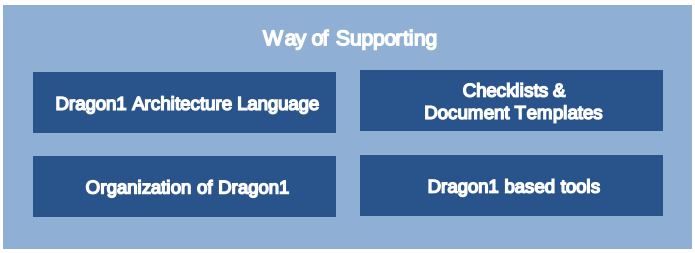

Figure 2.16. Way of supporting framework.

The model **Way of supporting framework** shows four 'way of supporting' concepts. When making architecture designs and determining the architectures, the architect in accordance with Dragon1 is supported in a specific way, i.e. by software tools, sample posters, templates, checklists, reference maps, generic symbols and certification.

2.4.5 Visual Enterprise Architecture

A new model emanates when building architecture concepts and a new way of looking at the business and information science assimilate. In Dragon1 this model represents visual enterprise architecture. This model is elaborated upon in Figure 2.17 and provides an example of how among others the architect as a designer of a structure translates function into form. Similarly, how the architect creates a design at different levels of abstraction following his knowledge and experience of concepts and stakeholder requirements and needs.

Dragon1 model – Visual Enterprise Architecture

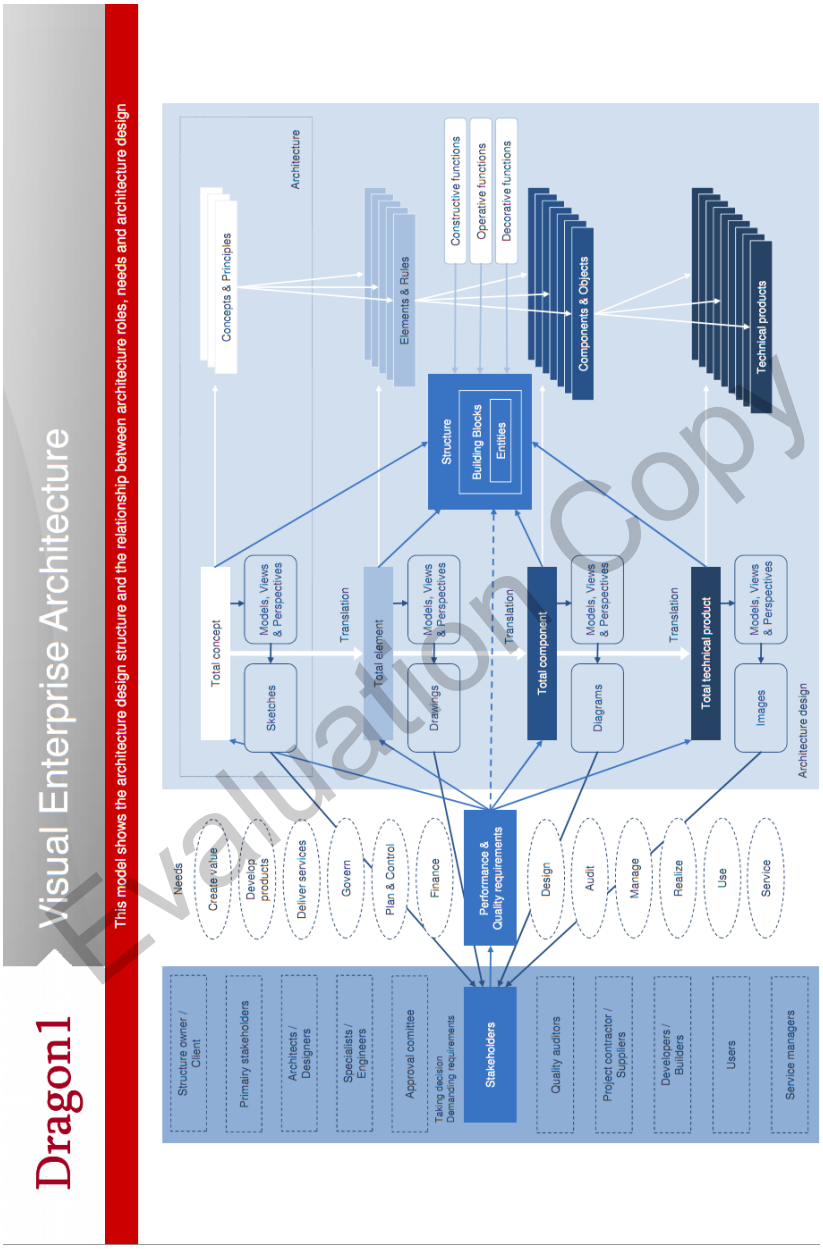

Figure 2.17. Dragon1 Visual Enterprise Architecture.

The model **Visual Enterprise Architecture** contains the following Dragon1 core aspects:

- Stakeholders of a structure have requirements regarding the quality and performance of a structure; and they have requirements regarding independent building blocks such as control, business, information facilities and IT infrastructure.

- The stakeholders present these requirements in support of their activities, such as their utilization, delivery, control, design, provision of resources, products and services, as well as the planning of the implementation of the structure or its pre-requisites. From activities, such as these, a need arises regarding (business) resources, (information) facilities and (IT) infrastructure. These resources and facilities will be applied to structures or part structures in the form of solutions, or even create new structures, which constitutes a facility in its own right.

- The many stakeholders involved in a structure share different points of view regarding its realization due to their different activities, needs, concerns or roles. This is why we often see opposing and seemingly irreconcilable requirements regarding performance or quality of a structure.

- The architect is obliged to think up solutions for a myriad of requirements. Sometimes, he is needs to design and test new concepts to successfully address conflicting opposites.

- The architect divides or differentiates requirements of a structure and abstract concepts underpinning this structure, the structure's logical functional requirements, physical or digital technical requirements of various component and objects, as well as vendor requirements regarding their provision of technical products.

- The architect divides the structure into constructive, operative and decorative functions and elements. Hereby the requirements pertaining to the performance of the construction elements prevail over the operative elements and decorative elements.

- The architect creates a decomposition of the structure at conceptual, logical and physical/digital level in support of concepts, elements, components, objects and technical products. By setting requirements, the stakeholders contribute in determining implicit design frameworks. The architect supports in setting the requirements, in setting the appropriate requirements and the correct formulation of the requirements.

- The architect assures that technical product requirement and supplier dependence is avoided. The way to do this is by including stakeholders to formulate element and concept requirements underpinning technical products and components.

- Stakeholders such as the client are often given informal sketches by the architect representing decomposition, scenario and choice of certain concepts and principles as well as rules pertaining to the total concept. As the client is often owner and principal, he will use these sketches to support decisions regarding the choice of certain requirements. The client may often choose particular concepts definitively, but not the way in which they are supported by elements and components.

- The architect may produce informal (cartoon) drawings for stakeholders such as enterprise directors and management, which contain a decomposition, scenario and choice of certain elements as well as rules pertaining to the total element. These stakeholders, such as directors and management are often professionals capable of deciding if something is plan-able, affordable, controllable, and manageable. They often take the position of member of a steering committee having the power to make decisions. They use the architect's informal drawings to make decisions in the choice of performance and quality requirements regarding elements..

- The architect often produces formal diagrams for stakeholders, such as specialists and quality assessors showing decomposition, scenario and choice for certain types of components and objects as well rules pertaining to of the total component. These stakeholders are often experts capable of assessing whether a solution is realizable, integrate-able and reliable. They take decisions regarding the implementation of a structure or design details on the basis of visualizations. As such, they support the choice of certain

techniques and technologies. Only formal diagrams and schematics may be used in compiling contract documents and specification sheets.

- The architect may also produce informal photographic images such as architecture photographs, artists' impressions, persona and collages to stakeholders like a board of directors, to clients, and to peers. The photographic images contain decomposition, scenario and choice for certain technical products as well as rules pertaining to the total technical product. The stakeholders may not be subject matter experts, but may enjoy many years experience instead and as such, they are experienced in assessing the feasibility of a solution. They may take emotional decisions based on visualizations of the pros and cons of a solution and whether or not to use a solution.

- In order to limit risks in implementing a structure as much as possible, the architect must include as many details as possible in the architecture design. As such, he will exercise more control on quality and performance.

- By using sketches, drawings and images, the architect will attempt to produce atmosphere, emotion, feeling and animation to a design. This creates a wider platform of support, as well as more and better requirements regarding the actual design. Eventually, the architect will achieve consensus of the architecture design.

- An architect will produce visualizations of models and views as well as perspectives of the total entities, such as the total concept and the total component. Visualizations are brought in line with the visual preferences of a viewer. The side view of perspectives are brought in line with the focal points of the viewer, realizing from which view point the stakeholders will view the structure.

- The challenge for architects is threefold: (1) to come up with concepts by which all requirements can be covered, (2) to translate concepts to elements to components to objects to technical products and (3) to produce useful requirements for stakeholders with the assistance of functional decomposition.

- An architect does not only include visualizations as part of a design, but he also includes visualizations in determining the program of requirements; visualizations are often used to improve and to clarify requirements. In reality designs often lead to details for a program of requirements.

With the model **Visual Enterprise Architecture** we conclude three main points: (1) Dragon1 supports translation from concept to element to component and object as well as to a technical product. (2) Dragon1 makes an enterprise an approachable structure. (3) Dragon1 supports the architect with the creation of target group related communicative visualizations. With visualizations of side views, perspectives and models, the architect is in a position to enable stakeholders to make decisions in setting requirements for performance as well as the quality of a structure.

2.4.6 Dragon1 Wiki

The Dragon1 Wiki can be found on the website http://www.dragon1.org/wiki. It offers in-depth details of the Dragon1 method and starts in fact where the study book finishes. The Dragon1 Wiki contains descriptions of Dragon1 concepts and links to extensive Dragon1 content among which, the term file. Dragon1 Wiki is the platform where users and specialists meet to exchange knowledge and experience about visual enterprise architecture.

2.4.7 Architecture language and set of symbols

Dragon1 has a specific vision regarding symbols, which are used to visualize entities such as types of concepts, elements, components and objects as well as entities such as types of principles, architectures, plans and models.

A symbol must as much as possible represent the archetypical image of an intuitive concept metaphor. A certain class, type and sort of principle must serve as additional sub archetypes in or placed on the lower left hand side within a symbol. We recognize this as a composite symbol. Examples of a composite symbol are symbols in a chain process, business process and work process in stead of just one symbol for a process. There are also symbols for customer requests, business objects, information object contact channels, information product as well as communication services.

Dragon1 provides a set of more than 250 basic archetype diagram shapes for conceptual, logical, and physical entities. This archetype set of symbols mainly contains composite symbols, which indicate types of concepts in order to make architecture visualizations as accurate as possible.

2.5 Open standards

2.5.1 A future-proof approach

Dragon1 is an open method, meaning that the users, under conditions of a transparent process, can contribute to the continuous development of Dragon1. Since enterprises and institutions innovate on a daily basis, the profession of the architect is equally progressive. Each year new concepts, principles and findings are added, whereby the methodical approach for architecture can be honed.

- *An Open Method is a method whereby users of the method exert influence and contribute to the continuing development of the method. The continuing development of the method is done on a transparent basis.*

Due to the fact that Dragon1 is an open method, the method exists in an environment that allows it to develop continuously whilst remaining continuously available to all users.

Dragon1 consists as an open method of a total of open standards. Currently the following open standards are available.

1. Architecture Principles Standard

2. Visual Language Standard

3. Architecture Visualization Standard

4. Enterprise Architecture Design Standard

5. Architecture Quality Management Standard

6. Architecture Stakeholder Management Standard

7. Architecture Communication Standard

8. Architecture Implementation Standard

9. Architecture Service Management Standard

10. Architecture Start-Up & Initiation Standard

11. Architecture Application Standard

12. Architecture Framework Standard

13. Architecture Documentation Standard

14. Architecture Maturity Standard

15. Architecture Enterprise Innovation Standard

16. Architecture Requirements Management Standard

17. Architecture Rules for Architects Standard

18. Solution Architecture Design Standard

19. Architecture in Projects Standard (PXA)

20. Agile Architecture Standard

21. Architecture Governance & Management

The Dragon1 Architecture Foundation accompanies the continuous development of Dragon1 by way of a process that allows a controlled manner for handling change requests, which assesses and provides information to Dragon1 of organizational development via a change proposal. The proposal is in turn used to improve Dragon1 where necessary.

However, this means does not mean that all requests and proposals are accepted on a 1-to-1 basis and that they are processed on the basis of open standards of methods. Some requests and proposals, which are target group specific, may for instance be included as best practice. This way, the method becomes generically available and usable for a very large group of architects for all types of enterprises and institutions.

Dragon1 model – Method as a movement

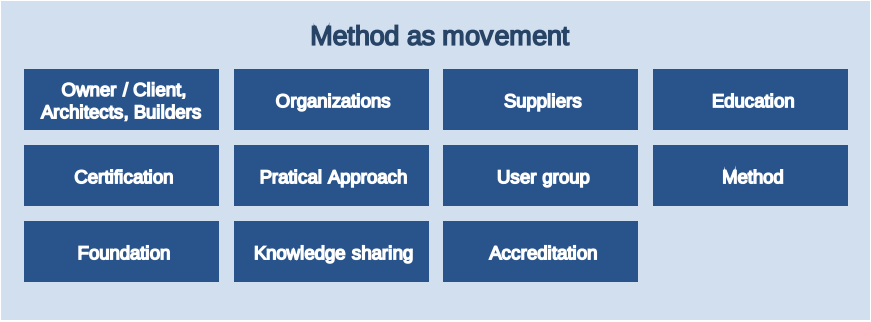

Figure 2.18. Method as a movement.

In the model **Method as a movement** we discover an example how different parties, as a movement, can play a role in Dragon1 using visual enterprise architecture in an innovative way.

2.5.2 Alternative approaches, theories en techniques

Holistic approach

Dragon1 reintroduces the enterprise holistic approach, stating that among others everything in an enterprise is integrated and everything constitutes all. For example, an employee always arrives at work early and keeps appointments. This approach has a holistic ripple effect regarding to higher profit margins of an enterprise and as such the continuing existence of that enterprise. The holistic principle states, that you are the enterprise. Dragon1 uses the holistic approach to create a supportive platform of all stakeholders involved in a structure. Everyone is important and everyone plays their role.

Contingency theory

Dragon1 embraces the contingency theory, which states that all systems are connected to each other. A change to one system leads to a change in another system whereby both systems retain a fragile balance.

Dragon1 states that each enterprise is connected and in contact with its surroundings: the other enterprises. Each enterprise wittingly or unwittingly provides a link to other enterprises, consequently every change in one enterprise requires evaluation of a chain of enterprises, for instance from a perspective of an enterprise's right to exist.

Alternative standards and techniques

Dragon1 uses standards and techniques to design structures. Below is an overview of standards and techniques which are recommended by Dragon1 to explore and use (copyright permitting). The theory of Dragon1 often uses these standards and techniques to make it possible to create high quality architectures and designs.

- Application modeling (OOA/OOD, UML, ERD, ORM, SOA, Web2.0-standards)
- Database modeling (BCNF, IDEF0)
- Process Modeling (BPEL, BPM, BPR, BPMN)

- System Development (SDM, Waterfall)
- Product development/Marketing (Shark)
- Project Management
- Information Management (BISL)
- IT Service Management
- Testing (TMap®)
- Controlling (Corporate Governance, COBIT)

Business Process Redesign (BPR)

Lateral thinking to an architect is a must to arrive at ideas and solutions for new concepts whereby an interpretation can be made of contradictory terms. Hereby BPR is an approach that assists the architect. We cover BPR from the list of techniques because it is of such importance to the architect that it should become a second nature.

The way in which an enterprise is automated depends largely on the extent in which the fundamental enterprise procedures are reconsidered. By essentially looking at fundamental issues in a different way, we soon reject supporting automation of the manual process and arrive at totally different formats. A different format often leads to new possibilities and to the use of new technologies.

> ■ *Business Process Redesign is the fundamental redesign of processes.*

BPR has a saying: 'if you automate a process with a problem, you automate the problem.' And that is exactly what should be avoided.

For instance, if a person wants to increase the rate by which letters are sent to a large number of customers informing them about the current status of their case in progress, BPR states that it is essential to look at 'why' letters are sent and whether it is still necessary. Are there perhaps other ways of communicating? Currently, the enterprise sends the letters themselves, but possibly this could be outsourced. In

short, the enterprise should utilize available possibilities and not walk around with blinkers as a result of the implementation of assessment processes and design phase. During the realization phase, a person should take account of the current facilities and the available time, competences and budget.

BPR should stimulate the architect by asking the question 'why?' Why does an enterprise work in a certain way? Why does it have certain products and services? Why are certain information systems built in a particular way? Why hasn't something been done before and so on. The answer to these questions help the architect to separate from the physical enterprise facilities on the road to to a vendor and technologically independent analysis of the enterprise at a conceptual level – in other words, enterprise architecture.

2.6 Questions

After studying this chapter the reader is able to answer the following questions.

2.1. Name two concepts from the way of thinking and describe the way they work, their function and the relationship between them.

2.2. Name two concepts from the way of working and describe the way they work, their function and the relationship between them.

2.3. Name two concepts from the way of representing and describe the way they work, their function and the relationship between them.

2.4. Name a distinctive difference between an open method and a non-open method.

2.5. Name a distinctive aspect of the Dragon1 set of symbols.

Chapter 3

Visual Enterprise Architecture in practice

The benefits of the new view on enterprise architecture.

3.1 Study Objectives

After completing this chapter, the reader is able to:

- state, how visual enterprise architecture can be applied as a management control instrument.
- state, how principles and concepts can be utilized regarding the challenges relating to a client's requirements, starting points and pre-conditions.
- state, which role quality, quality requirement and integral solution have within visual enterprise architecture.

3.2 Introduction

The chapter 'Visual Enterprise Architecture in practice' elaborates on how visual enterprise architecture is included within enterprises, as a management control instrument to enable board, directors and management to provide solutions to enterprise issues. Regarding these issues, we distinguish challenges and focal points as well as strategic starting points and principles. We will also discover how visual enterprise architecture can be integrated within an existing quality system and within an enterprise planning & control cycle.

In this chapter we pay attention to a number of common management concepts, such as ambition, strategic intent, concern, strategic starting point, goal, target and activity. We also focus on how management concepts are inter-related with and within visual enterprise architecture.

While explaining these management concepts, it becomes clear how visual enterprise architecture creates much added value in confronting challenges, and finally in realizing enterprise aims and objectives.

The questions we aim to answer in this chapter are: What is the added value of visual enterprise architecture above and beyond the present use of enterprise architecture? What problems can be solved by visual enterprise architecture and how is visual enterprise architecture used in solving different issues?

3.3 Visual enterprise architecture: the management control instrument

Dragon1 champions visual enterprise architecture as a strategic management control instrument to solve enterprise issues. Visual enterprise architecture enables enterprise architecture to be navigable at a strategic level, planable at a tactical level and implementable at an operational level.

- *A strategic control instrument is an aid used by board, directors and management to direct, with the use of frameworks and perspective, a controlled execution of enterprise activities to realize enterprise objectives.*

In many enterprises we all too often notice, at the start of a change initiative or program that certain strategic decisions were not being taken. Or that, due to progressive knowledge, certain far-reaching choices have to be made. Visual enterprise architecture displays the impact of certain change initiatives, how show stoppers and high risks can be avoided by visualising of realization -, migration - and implementation scenarios. This helps enterprises to implement complex strategic changes in a controlled and phased manner.

Visual enterprise architecture shows which concepts and principles have to be included by an enterprise or need te be included to implement a certain mission, vision and strategy in order to realize objectives. Regarding this, an architect produces different types of visualizations of different models, views and perspectives of a structure, which are exactly in line with the different stakes of different groups of stakeholders.

Visual enterprise architecture increases, as such, the added value of enterprise architecture for directors and management, which enables the making of strategic choices and decisions on the basis of inter-related strategic starting points, concepts, principles and change initiatives.

- *Visual enterprise architecture is the art and science of creating architecture designs for structures with a focus on visualizing different models, views and perspectives. This enables clients, stakeholders and engineers to increase management control, have insight and overview, exercise support and conduct the realization of the design, functionality, performance and quality of structures.*

3.3.1 Visual enterprise architecture business case

Enterprises interested in improving the quality of their goods, products, services and business processes increasingly use architecture to achieve their aims. For instance, an enterprise intends to sell their products to more customers at a lower cost to accomplish a higher turnover and profit in the future. The use of architecture currently takes place with variable results.

Enterprise programs and projects are increasingly supplied with project start architecture (PSA), which is a kind of solution architecture befitting an enterprise architecture. In a restructuring program a business process or IT system, or a combination, will be designed, realized and implemented. In this case the purpose of the PSA would then, for instance be increasing the ability to realize the process or system that is being developed and eventually its suitability within the enterprise.

Often these PSA's only provide guidance. They consist of voluminous text documents and are difficult to comprehend, subsequently they are seldom used.

- *A project start architecture (PSA) is a document in which reference architecture, enterprise architecture and solution architecture come together in a project – they are described as a guiding framework to make project decisions.*

The risk becomes irresponsibly high when, over the length and breadth of an enterprise, parts are suddenly removed or renewed. When this happens without sufficient integral insight and overview of the current core activities model, chances are that a 'supporting wall' is displaced wiht the consequence that a part of the operation, the scope of work, or the implementation of activities will collapse. The collaps damage being bigger it affects the integral interdependencies, supporting structures within the enterprise, the operational entities and coherent support of the businesses within the enterprise.

If it is unknown at the beginning of a change how an enterprise operates as a whole, then it will also not be known if the intended adaptation will become an improvement or a deterioration to the enterprise. In other words, if the wrong supporting pylon of a bridge is removed, the entire bridge will collapse. Therefore, it is of major importance that sufficient insight and overview is present when starting large scale restructuring within the enterprise.

The business case of visual enterprise architecture covers this item thoroughly. The business case proves that it is the visual enterprise architecture that supports

innovation at a responsible level of risk. Visual enterprise architecture makes it possible to control the level of innovative risk, because among other aspects, at the drawing board the different points of view of the stakeholder are incorporated in detail into the structure. At the drawing board, any show stopper will be discovered, and can then be eliminated for relatively little cost.

As long as visual enterprise architecture activities deliver what has been sketched by a business case, the activity can be considered advantageous. Reality teaches us that too many enterprises achieve too little or no result at all from their architectural endeavors, or have no appreciation of visual enterprise architecture for their business.

In the following paragraphs, a number of important management concepts at strategic level are elaborated upon, such as: mission, vision, aims and strategy, which all play an important role regarding visual enterprise architecture.

- *A visual enterprise architecture business case enables enterprises at a responsible level of risk, to remove, optimize or renew parts within a shorter period of time at lower cost. In other words: with visual enterprise architecture renovation and innovation can be designed and realized at a higher level of quality and with enhanced performance.*

3.3.2 Challenges

The challenges of the enterprise can be translated into ambitions or strategic starting points, intentions and objectives. These challenges represent the enterprises' objectives. Challenges are used to assess how ambitious objectives can be pursued. A challenge also often indicates potential difficulties in realizing enterprise objectives.

Examples of challenges would be: 'to increase market share in a saturated market' or 'to produce more efficiently and more environmental friendly without increasing the price to the customer.'

Examples of challenges which enterprises are prepared to face, but often fail to achieve are: mergers and cutting costs without forced redundancies, to become the top company in a certain market, realize certain customer satisfaction ratings, as well as to become independent of every possible supplier.

- *A challenge is an exercise or objective that is achieved against great odds. Often a challenge can*

be unrealistic or the bar is set too high.

Challenges like mission, vision and identity, are an input when formulating strategic starting points, strategic intentions and ambitions.

If ambitions are to be realized, they must be specified in terms of concrete objectives.

- *An ambition signifies realization of a higher objective or signifies targeting a higher objective.*

An enterprise is obliged to adopt certain strategic intentions in order to realize its ambitions.

- *A strategic intention is the expression of an enterprise's intention regarding its strategy. It constitutes the will to realize ambition, goal or objective.*

Too often enterprise challenges are not translated into policy, nor into appropriate enterprise solutions. If an enterprise challenge entails production in an animal friendly manner, whilst making sufficient profit, visual enterprise architecture is the appropriate instrument to design and select realistic production methods.

Supported by visual enterprise architecture, board and directors are enabled to manage development and direct the way of working within an enterprise taking into account recommended and mandatory strategic solutions.

Often goals and targets are not derived from an enterprise's mission. However, the moment an enterprise does derive goals and objectives from its mission, every business activity becomes legitimized proportional to the level of quality attained and effort made. Visual enterprise architecture ensures that on the one hand the importance of a carefully formulated mission statement is visualized and on the other hand the cohesion between objectives, activities and quality of goods is shown clearly and in an overseeable manner.

- *A objective is the underlying reason why certain activities are carried out or goals to be realized.*

- *A goal is a requirement to realize a situation, it is derived from an enterprise's mission statement and identifies the entity required to achieve the goal. Goals can be found at all*

enterprise levels such as enterprise goal, business goal, business functionality goal, business process goal and department goal. Perpendicular to this set up, we see the classification of strategic goal, tactical goal and operational goal.

- *A target is a concrete and detailed formulated (part of) an goal.*
  A target should be as much as possible formulated in accordance with 'S.M.A.R.T.'-principles.

3.3.3 Concerns

Stakeholders have concerns, which they wish to see incorporated in problem solutions. These solutions must be visible during development in the result. These concerns come as a result of the role, position or task, the stakeholder has in relation to the enterprise.

Concerns can be:

- The financial manager, who is obliged to keep an eye on a certain financial ratio.
- The IT manager, who is obliged to assure the integrity of an IT solution.
- The facility manager, who must keep an eye on the use of certain products and services.
- The HR manager, who must assure the search for new essential competences.

The stakeholder always checks whether the goals and objectives include focal points regarding his areas of concern. For a better realization of objectives, visual enterprise architecture can make clear which solutions can be addressed regarding the various focal points.

Example: An architecture visualization is made for an HR manager, which shows a competence perspective for the current and future situation. These perspectives show an internal shortage of resources and competences and that if no action is taken, how much external contracting will be required. These perspectives also make it clear to the HR managers which competences are covered by which functions and how much the enterprise is required to do regarding competency-based retraining,

further training and refresher training. By visualizing this, the complexity of the issue becomes comprehensible, and thus it becomes a common operating visualization to various people and also how this will promote faster and better decisions.

- *A concern is an aspect or subject relating to someone's area of concern regarding his portfolio of responsibilities.*

3.3.4 Strategic starting points

Normally an architect receives an assignment from the board or directors on the basis of a program initiative or request to produce an architecture design showing the envisaged changes. The architect will produce architectural designs of integral business IT solutions. These solutions fit within the strategy of the enterprise, next to the set assignment requirements, in other words they will comply with the overall quality requirements.

In order to include the quality requirements into the architectural design the architect uses management concepts such as ambitions, strategic intentions and strategic starting points, in which mission, vision, philosophy, culture and identity of the enterprise are reflected.

Strategic starting points are a more difficult subject compared to ambitions and strategic intentions, because strategic starting points tend to generate more impact. If ambitions and strategic intentions describe the enterprise's aspirations, then strategic starting points usually cover everything and everybody that needs to be taken into account in and around the enterprise.

An example of a strategic IT starting point is when an architect produces an architecture design that incorporates statements such as: 'We will be, we are, and we remain the best IT provider for our internal customer'. This is a strategic starting point that is often used if the internal IT department is not the only supplier of IT services to the enterprise.

In order to be able to work effectively with strategic starting points, the criteria mentioned below are always applicable.

- A strategic starting point must be constructed around enterprise entities and the quality aspects applied to the enterprise, which are: the best services and the safest products, etc.

- A strategic starting point must be formulated in the language the client understands. The client is the architect's customer and also often the owner of a domain such as a business function.
- A strategic starting point must provide direction if it is to be used to change the enterprise. A strategic starting point must therefore be related to the enterprise entities.
- A strategic starting point must excite and stimulate, be descriptive and communicative. It must conjure up a visualization.
- A strategic starting point must as much as possible be formulated in a positive sense. Such as: 'the world changes and we change with the world'; and not: 'because the world changes, we have no option but to change along with it.' Or, such as: 'we provide the best products in the market'; and not: 'our customers are unable to make a choice on their own and therefore we assist them.'
- A strategic starting point often works in relationship with an ambition or a strategic intention for instance, an ambition could be: 'We are going to be the best in Europe,' could quite easily lead to a strategic starting point: 'We are the purveyors of the best products in the market because we are the technology leader in our field', which can then be communicated to customers. Simultaneously we could declare a strategic intention, such as 'we will invest heavily in our production process to deliver the best products.'
- A strategic starting point can also contain a certain trend, industrial development, society or economy, or a situation in the market or with a customer. Regarding these themes we can establish this position or set a goal. Examples of such strategic starting points are:
    - Our customers are becoming more articulate;
    - We only cooperate with parties pursuing our level of quality;
    - Our enterprise has got our basics right;
    - Nanotechnology is going to make it;
    - No enterprise can exist without a website or mobile telephony.

- *A strategic starting point is a declaration that makes it clear via its board and directors what the enterprise stands for, where it is coming from, and what it regards to be its basic and fundamental principles. A strategic starting point makes it clear what to take into account regarding, environment, situation, issues, trends, development and information in relation to the execution of its core activities.*

  A strategic starting point gives direction and imposes borders. As such, it provides structure, tranquility, rhythm and space. Strategic starting points have a great impact in setting and implementing strategy. A strategic starting point resonates environment, ambition, objectives, intention, critical success factors, mission, vision, philosophy, culture as well as the identity of the enterprise. Strategic starting points are preferably formulated as statements.

Using visualizations such as sketches, drawings and images the architect depicts the strategic starting points of the various groups of stakeholders. Also, how all enterprise developments and solutions can be inter-related to these strategic starting points.

To have a clear view of the strategic starting points is of great importance. They constitute partly the point of departure and support the necessity of certain concepts and principles as entities of architecture. In fact, having a strategic starting point takes care of spatial direction or boundaries for realization of certain needs. These needs implicitly require the presence of a concept, or a principle that will be incorporated into the realization of a structure in line with strategic starting points.

Dragon1 model – Strategy, Architecture and Policy

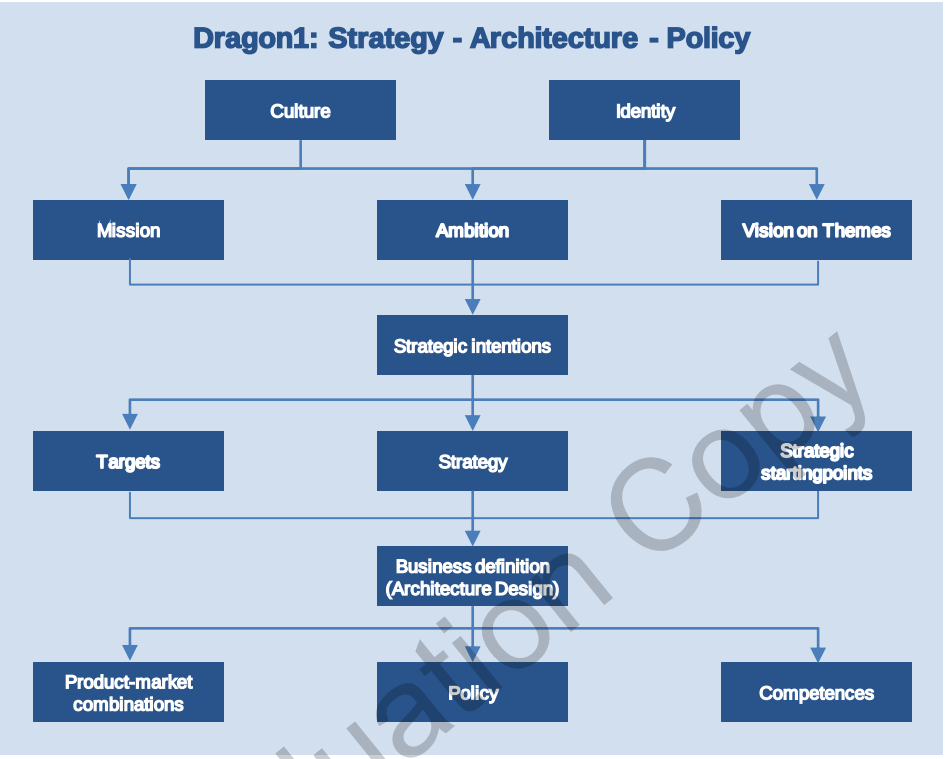

Figure 3.1.Relationships between different management concepts.

The **Relationships between different management concepts model** provides an example, on the basis of strategic starting points, how space and limitations shown in activity frameworks provide support to realize aims and objectives. The objectives are derived from ambition, whereby in respect to realizing this ambition certain strategic intentions are articulated. These strategic intentions result in specific enterprise objectives, activities and challenges. Strategic starting points and strategic intentions serve to formulate objectives and serve as a basis for determining the required or mandatory activities. Without the consistent and consequent use of management concepts as shown in Figure 3.1 an enterprise has insufficient context from which it can operate.

The management concepts in Figure 3.1 are formulated only in terms of quality aspects, such as security, sustainability and realizability and in terms of performance aspects, such as 24x7 availability, 100% digital and 0% waste.

Please note, in Figure 3.1 management concepts such as mission, thematic vision, culture, philosophy and critical success factors have been omitted. These aspects are covered in the model Identity with a link to strategy. Please refer for more information to http://wiki.dragon1.org.

The architect's task as conductor forming images is to deliver an image to the board and directors and to ensure that it is a clear and transparent total of the challenges, strategic starting points and ambitions. This is a pre-conditional to successfully provide form to enterprise innovations with the support of visual enterprise architecture.

The mission, vision, philosophy, culture and identity of the enterprise must be used by the architect, the board and the directors to arrive at a unique total entity of these challenges, strategic starting points and ambitions. When formulating this total entity, the architect must ensure that the quality aspects are foreseen, such as controllability, transparency and customer focus. This will give structure to the total. On a tactical and operational level the model shown in Figure 3.1 also provides business and operational objectives.

3.3.5 Principle statements

On the whole, enterprise stakeholders formulate and reform requirements, objectives, rules, pre-conditions, policy, strategic intentions, core values and critical success factors, and many other important elements.

Practicable architecture teaches us that these increasingly important statements, 'principle statements' or 'propositions' are labeled as 'principles,' however they are not. The principle statements are, however, so important that they ought to be reformulated into real principles.

Examples of principle statements

- The customer is king.
- With us, the customer is never in the wrong place.
- We work in a process oriented way.
- Customer data is protected by our website self service applications.
- We maintain customer data and keep it as original as possible.

- Without structure and standardization, only chaos ensues.
- All software applications are accessible and are inter-communicable via web services.

Such statements are in conformance with the theory about strategic starting points; also they can be used as strategic starting points. Because, as far as the board and directors are concerned, each and everyone ought to take these statements into account, although it does not mean that these principle statements are principles.

None of the principle statements describe a maintained interaction with a certain result produced by the interaction. The principle statements do, however, all address an underlying principle, which would be preferred or taken into account. By considering these statements as soliciting titles for principles, and to restructure them into a maintained activity, we arrive at possible intended principles.

Hypothetical examples of restructured principle statements into short statements of correctly formulated principles are:

1. **By** treating a customer as king, as part of an internally agreed approach, and to make him feel that he is always right, an enterprise ensures that the customer is very satisfied with the service and products and that as such, he will remain a customer for a longer period of time and will speak about the enterprise in a positive manner.
2. **By** never turning a customer away when he contacts the enterprise with a customer query from a multi channel front office perspective, it is assured that more products become available and service time is shortened, by which the internal efficiency and external customer satisfaction are increased.
3. **By** working in a process oriented way and assuring that the industry standard regarding process structure and process set-up is maintained the enterprise assures that every one inside and outside the enterprise becomes aware that this is the way to work. The effect will be that the enterprise will work more efficiently and that changes can be incorporated faster. This way the allocation of ownership also becomes simpler. In turn this assures that it is possible for enterprise activities to become better managed, controlled and adjusted when necessary.

4. **By** always maintaining strict supervision and control, that only https connections are used, it is assured that data is shared with customers over the internet in a secure way, whereby customer privacy is guaranteed.

5. **By** having a strict control of the input and maintenance of customer data and that as such input of information takes place only once, it is assured that inconsistent double entry of the same data is minimized if not totally avoided.

Here, each principle belongs to a concept. The name of a principle is derived by naming the concept and an activity or benefit of the concept that is described together with the principle. For the first example principle, it could be the longstanding relationship principle of the 'customer-is-king' concept.' If a principle contains an entire core concept, we often refer to it with a name, such as 'the-customer-is-king' principle. Providing a principle with a name ensures that we are forced to think about the concept used to describe the principle. This is an aid to reconstruct principle statements into real principles in an even better way.

When these principles apply or are going to be applied in an enterprise, they are always given with an additional type name linked to the domain or system in which they are applied, next to the type name which they carry from the concept. For instance, when example principle four has not brought into relation with the enterprise, then it is an 'information security' principle of the concept 'communication security of internet data'.

This example principle about secure connections is for example an 'information security reality' principle within an enterprise if it has already been applied as such. If the example principle has yet to be applied, it is an 'information security design' principle. It depends on the situation which characterization is given to a principle or by which characterization we name it and what we are going to use it for. The same principle can as such have or be given different names and characterizations for different objectives.

3.4 Quality & Quality aspects

In the previous paragraph we briefly mentioned how quality aspects constitute an important part of strategic starting points. But which role do quality aspects play in visual enterprise architecture exactly?

3.4.1 The goals of quality aspects in visual enterprise architecture

Each concept has a certain action resulting in a certain qualitative outcome. Therefore quality is directly linked to visual enterprise architecture. Within the architecture the architect chooses a safe solution allowing a certain degree of security (i.e. a quality aspect) to be incorporated within the possible solution.

- *A quality aspect denotes the dexterity of a system. A quality aspect is a non functional situation or non functional performance for which an objective is formulated.*

Please note: a quality requirement is a requirement that has been incorporated into a quality aspect. We must be careful not to confuse quality requirements with performance requirements of a system.

Dragon1 follows ISO9126, which denotes a generic list of quality aspects.

For instance, the quality aspects in the ISO9126 model can be used for each business process, information system and computer network as part of a program of requirements on the basis of experience or as the requirement set at the beginning of a project.

If we look at the day-to-day activities of enterprises in general we see, for instance, that in relation to primary business processes, information systems, applications, information gathering, client computers and servers, quality aspects are not always formulated clearly and transparently. A description and visualization of perspectives for each quality aspect of the named entity alone will make an enterprise better manageable many times over.

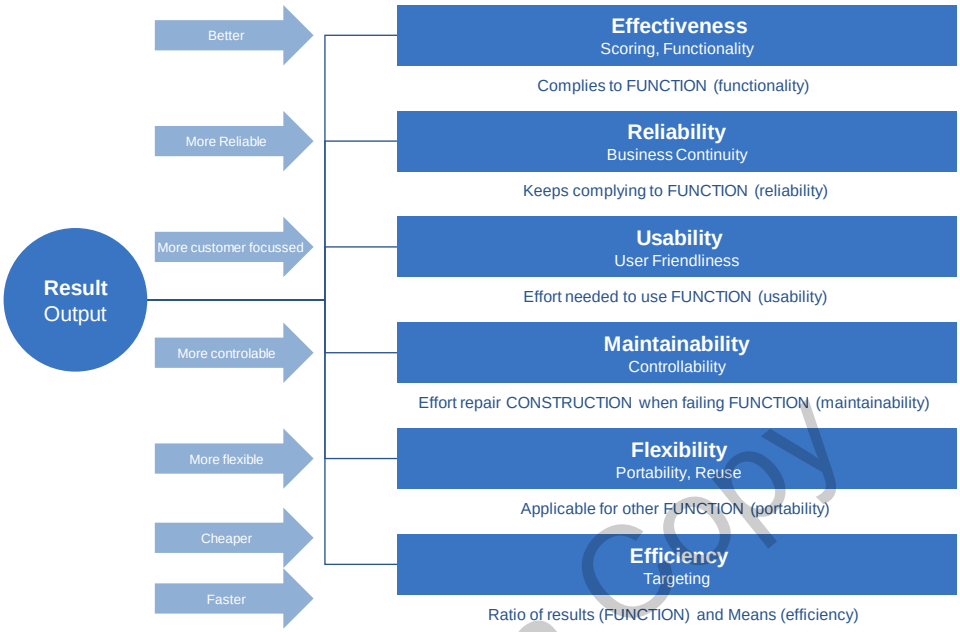

Figure 3.2. Example of ISO9126.

By visualizing the architecture of a solution and to relate this or map this to the set of quality requirements, it can be made clear at the drawing board how well the solution has been designed.

Say that the IT department of an enterprise holds two customer databases and has designed a solution to import data from one customer database to the other database, so that there cannot be any inconsistency between the two databases. A customer might only exists in one database at any one time.

Business management provides an architect with an assignment to assess where a problem might arrise if a new solution is chosen which includes the removal of a certain product group and its associated customers.

The architect comes across the customer database and discovers by way of a visualization exercise that it is impossible to make a simple distinction between the potential customers of a certain product group. The architect is now able to show that the IT solution is optimal, but taken from a business point of view, will cause an obstacle in removing the product group quickly.

With sketches, drawings, diagrams and images of the various perspectives of concepts and principles, the quality aspects of the solution will become visible. The

design and realization within visual enterprise architecture enables an increase of quality and performance delivered by the functions of the system.

3.4.2 Quality models

Nowadays, all enterprises employ a specialist whose job it is to focus on the visualization and manageability of the quality of developments, production, processes and services. However this does not happen successfully everywhere, but it should and it therefore allows visual enterprise architecture to become the chosen alternative approach to increase integral quality.

Every enterprise prefers absolute quality control of its products, services and processes. As such, it is possible to determine quality requirements for certain quality aspects, and to agree to mandatory requirements concerning delivery and effort between customer and supplier.

A model that assists well in ensuring the delivery of continuous quality of products services and processes is the International Quality model.

International Quality-model

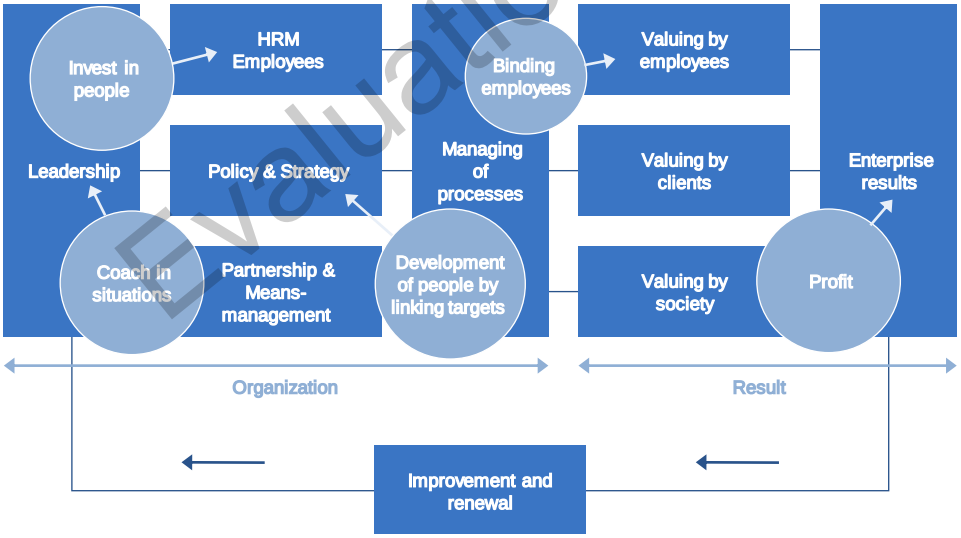

Figure 3.3. Example of International Quality model.

In public and non-profit organizations the International Quality Model is used more oftenly. The International Quality Model is made of nine parts, which cover an enterprise from a process point of view. Since the environment is included, the

International Quality Model is useful for enterprises and institutions that are linked to others especially in the care, education and public authorities.

This model is especially useful in visual enterprise architecture because it is a generic and understandable model concerning enterprises. International Quality Model considers an enterprise as a system that is continuously interacting with its environment. Dragon1 underpins the division and systems operations in enterprises as depicted by International Quality Model.

By visualizing architecture it becomes clear what is available to an enterprise to reorganize itself. Architecture is regarded literally as 'the supporting walls and the infrastructural facilities of a building'. The supporting walls and infrastructural provision of a structure do not change if a living room and a bedroom are inter-changed. However, to inter-change a bathroom and a living room requires adaptations of supporting wall and other provisions in the building.

Visual enterprise architecture contributes to the International Quality Model through quality because it inter-relates and prioritizes requirements and objectives of all stakeholders. Following this, it provides visualizations to show to which extent concepts and principles cover quality requirements to realize the objectives.

Thus, implementing an enterprise quality model assures better adherence to agreements within the enterprise.

- *Quality of an enterprise is the degree by which production of goods and products, service delivery and business process implementation comply with previously set quality requirements.*

3.4.3 Enterprise planning and control

One of the most important activities of an enterprise is to monitor, adjust and continuously improve the quality of products, processes and service. This is achieved by planning and control processes. If an enterprise uses a planning and control process we notice that such an enterprise is able to incorporate changes at the right moment at the right time. The reason is that an enterprise, or parts of an enterprise, require time to react to change. On the other hand we also notice that in the event an enterprise lacks a proper planning en control process that these enterprises are often late to react.

Say, an enterprise IT director wishes to eliminate an obsolete IT system, because the maintenance costs are too high, and that recently a new product had come on the

market that would replace the obsolete IT system perfectly. The commercial director wishes to avoid any conflict with the IT director and proposes to phase out the obsolete IT system in a year's time. The IT director is, however aware that a new version of the old product has been planned to hit the market. Without a proper planning and control process, the enterprise will have difficulty to replace its obsolete IT system.

In support of the above, Dragon1 has developed a reference model that can be used as an enterprise Planning and Control model.

Dragon1 model – Planning and Control

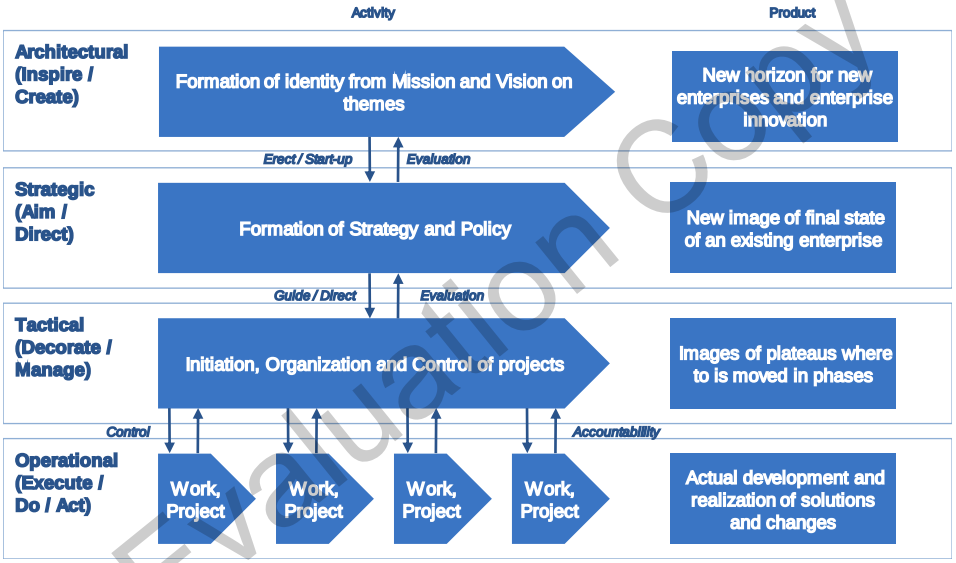

Figure 3.4. Planning and Control.

In the **Planning and Control model** the realization of the intended objectives is shown at a strategic, tactical and operational level. These objectives are translated into required interventions and changes enabling the realization of these objectives. By setting agreements between the different enterprise levels, the enterprise is able to carry out changes and interventions that fall within the above mentioned model.

A new horizon is created by establishing identity. This process constitutes input for exploring that horizon at strategic level more closely. By doing so a new situation becomes clear. At tactical level, working toward a new situation is shown in a step-by-step manner. In turn, this is translated at operational level to the implementation of projects related to developing or realizing plans and innovation.

It is necessary for the architect to be aware of how the architecture is linked to the planning and control cycle in order to avoid interference with that cycle. If a planning and control cycle is not incorporated, the chances are that at a lower enterprise level solutions do not fit within the accepted frameworks, causing the strategy to break.

Sometimes short term solutions appear useful, however in the long term they are not. The evaluation to choose for a short or long term solution can only be made at a strategic and tactical enterprise level.

Say, an IT department provides IT services to business departments. The IT department employs a few IT workers with extensive knowledge about the IT 'Virtualization of Computers' trend. The trend entails that one single physical computer presents itself as a cluster of a hundred computers at the same time. However, if one single computer has to perform simular to one hundred computers, it can be expected that it has to cope with a much higher load.

At its own initative, the IT department embarks on a virtualization program to replace 100 physical computers with 100 virtual computers upon a single computer. From a management perspective, this appears to be a clever idea, since system administration can be performed by less people and less physical space will be required. In fact there are some more additional advantages; however what the IT department was not aware of is that the directors had planned additional business departments, whereby it would be better to install 100 physical computers with far more capacity than 100 virtual computers.

An architect becomes invaluable when the planning and control cycle needs to be made clear by means of visualization.

3.5 Questions

After studying this chapter the reader is able to answer the following questions.

3.1 How can visual enterprise architecture be used as a management control instrument?

3.2 How are principles and concepts related to client requirements, starting points and pre-conditions?

3.3 Indicate which role is allocated to quality, quality requirements, and integral solutions within visual enterprise architecture?

Chapter 4

Dragon1 Way of Thinking

The foundations of visual enterprise architecture.

4.1 Study Objectives

After studying this chapter the reader is able:

- to describe which concepts constitute the Dragon1 Way of Thinking.
- to describe which types of concepts and principles there are.
- to give an example of a concept and an example of a reality principle.
- to describe the types of entities present in a structure.

4.2 Introduction

The chapter Dragon1 'Way of Thinking' elaborates on the first part of Dragon1. In the 'way of thinking' of Dragon1, the focus is on the theoretical foundations of the method: how does Dragon1 help the architect to observe enterprises and architecture in a different way, in order to display its added value.

The starting point of Dragon1 presents a way of thinking required to enable the architect to design architectures such as a reference architecture, with the ultimate goal to solve challenges within the enterprise. This enables the architect to accept an architecture design assignment to design an enterprise structure, assuring high quality and conforming to performance requirements of the structure.

This chapter explains which types of architectures Dragon1 acknowledges within an enterprise and which specific role concepts, principles and style elements play within visual enterprise architecture. The focus of this chapter is more concerned with the 'why' of architecture, architecture design, concepts, principles and style elements than on the 'what' and the' how'. This chapter helps the architect on his way to form his own vision of visual enterprise architecture.

The following paragraph sketches which Dragon1 'way of thinking' concepts constitute an important part in understanding visual enterprise architecture. The 'way of thinking' concepts are presented within 'the way of thinking' framework. It poses the question why Dragon1 is constructed from these 'way of thinking' concepts,' and what exactly are the inter-relations between these 'way of thinking' concepts.

4.3.1. Way of Thinking framework

The Dragon1 'way of thinking' can be described within one single framework. The framework consists of eight 'way of thinking' concepts. The 'way of thinking' specifically questions: why the architect uses concepts, principles, meta models, the human measure, strategic starting points, mission, vision and policy in developing architecture designs and architectures to create sustainable and future proof structures.

Dragon1 model – Way of Thinking Framework

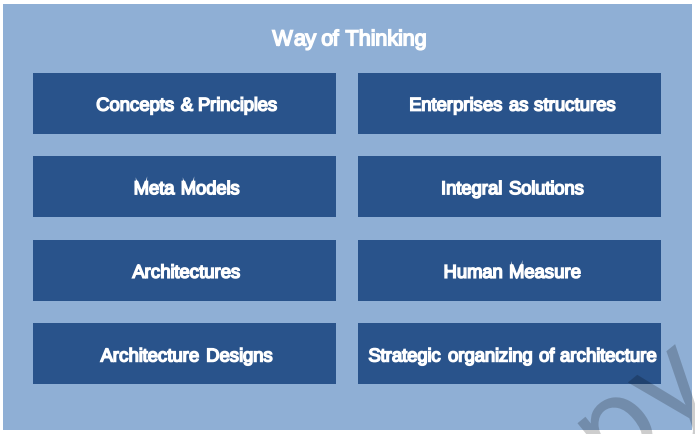

Figure 4.1. Way of Thinking Framework.

The 'Concepts & Principles' concept denotes the architect's advantages in terms of working with concepts and the reasons why. How does an architect select the appropriate types of concepts to work with? How does the concept work and, where can the concept be used? Also how does an architect recognize and formulate principles which are subject to a primary concept, and use this as a quality requirement to make a concept work.

The concept 'Enterprises as structures' shows how an architect can visualize an enterprise as a structure AND that visual enterprise architecture is an instrument for creating architecture designs of structures, whereby the structure's requirements are first translated into concepts, than into logical elements, and thereafter into components all responsible for delivering the required performance. By using this method an ingenious structure appears, which is sustainable and future proof, attractive and robust whilst performing in an optimal, functional manner.

The concept 'Meta Models' elaborates on the usefulness and necessity for an architect to use meta models in order to create architecture designs. What are the views and perspectives of a structure and what advantages are there for the architect to work with different model views and perspectives?

The concept 'Integral Solutions' denotes that organizational solutions increasingly form a coherent set of business processes and information systems they influence all business functions and business divisions in the enterprise such as production, purchasing, finance, HR and IT. Consequently, solutions must be designed in an

integral manner taking into account the solutions environment. This contributes to a sustainable, future proof and optimally functional solution.

The concept 'Architectures' describes the different sort of architectures present in an enterprise. The architect can create and describe these architectures, or can even retrospectively create these architectures, but never the less use the structure's advantages to make future improvements to the structure.

The concept 'Human Measure' points to the usefulness and operability of these structures. Ultimately it is the user that have to work with them. How can the architect to take this into account in the best possible way?

The concept 'Architecture Designs' shows why the architect creates architecture designs. What are architecture designs and how are they put together? What types of architecture designs are recognized?

In the concept 'Strategic organization of architecture' we learn how at all levels of the enterprise, working with architecture can be embedded. Organizing and managing architecture at an appropriate level enables an enterprise to avoid making the wrong decisions at the wrong time.

In the next paragraphs we subsequently focus on a number of framework 'way of thinking' concepts at all levels, from top to bottom and from left to right. Each way of thinking concept is elaborated upon by one or more Dragon1 models.

4.3 Concepts

This paragraph explains why concepts are so important to the architect. Dragon1 makes it clear what concepts are and which types of concepts the architect can distinguish.

By being able to distinguish concepts it becomes easier for the architect to find solutions for conflicting requirements and wishes and to incorporate them into a structure.

For the sake of understanding we repeat once more the definition of a concept:

- *A concept is an approach, an abstraction of an entity or an idea. A concept is often used as a directive to a solution.*

Example of the use of concepts

'Dear Architect, I would like to sell my products via the internet to customers who are able to change their customer details themselves, as well as monitor the status of the delivery of their ordered products

'Dear client, we don't know exactly how we are going to run it, but I propose that we combine the concept of a 'drupal webshop' with a 'web based self service' concept and an 'online parcel tracking' and 'tracing service' solution. We will then integrate everything on the basis of web services.'

This example clearly shows how the architect, at a conceptual level can sketch a reasonably concrete image of a solution by means of different types of concepts.

Many details are left out and there is a lot of freedom of choice for the customer and architect regarding the performance and quality of the structure. Only the choice for the stated concepts secures a certain quality: the possibility to sell via the internet, the management of a person's own personal details and the monitoring of the delivery status.

4.3.1. Overview of concept types

Viewed from a Dragon1 perspective, the world exists from different types of concepts: Constructive concepts, operative concepts and decorative concepts.

In an enterprise three disciplines are prominently present: business administration, informatics and IT. Within these three disciplines, all three types of concepts exist; as such business administration includes constructive concepts, operative concepts and decorative concepts. The same applies to informatics and IT.

A constructive concept is a concept that adds robustness and resilience to a structure, for instance, the separation of an enterprise into business functions. An operative concept is a concept that adds implementation of action and activities to a structure. For instance, remote collaboration in an enterprise. A decorative concept is a concept that adds atmosphere, emotion, feeling and experience to a structure. For instance, the branding of the enterprise, the image and advertisements are all decorative concepts.

An example of an operational business administrative concept is: managing personal customer details via a supplier website. Examples of decorative informatics concepts are information centers (the information columns and TV wall screens with news) at the counter lobby of a municipality showing information about the status of projects in certain areas. An example of a constructive IT concept is: virtual private network environments such as a production environment or a development environment.

Enterprise architecture, business architecture, information architecture and technical architecture consist, as we see later, mainly of business administrative, informatics and IT concepts. All these concepts are used by the architect as accurately as possible, assessed and documented. The purpose of this is to present them as existing proven concepts, in support of a solution for a structure. Later on, they are used in combination with new innovative concepts in support of designs of an innovative structure.

When the architect looks at the decoration of a structure he will endeavor to experience a certain emotion, atmosphere and feeling by using concepts, the architect can choose concepts with an appropriate quality and performance resulting in a required atmosphere, emotion, feeling and experience.

For instance, if an architect has to assure that everyone always works in the same way, he will then choose concepts relating to standardization and process oriented working instead of concepts such as self-managed teams with an individualistic and bespoke working approach. Even if the same can be achieved by all these concepts, standardization and process oriented working ‘are experienced differently’ than the concept self-managing and bespoke activities.

Fundamental, technological and supplier/product-dependent concepts

Dragon1 recognizes that in constructive concepts, operative concepts and decorative concepts there are three different levels: the fundamental, technological and supplier/product-dependent level. These conceptual levels can be subdivided in three ways: 1) global or generic, 2) average and 3) detailed or specific.

By creating distinction between different concepts, the application or the choice of concepts, as well as the classification of concepts becomes easier. This enables the architect to make choices that are quicker and better and result in sustainable and future proof solutions or concepts on behalf of the clients.

The classification of concepts in types and levels makes it clear which concepts belong together and which do not. By using this the architect has a wide choice of possibilities; which concepts to use and why. As a result, the architect is in a better position to determine whether an architecture consisting of concepts is dependent on a certain technology, and whether it is dependent on certain supplier products, or not.

By the same token, the architect sometimes needs to create new concepts based on existing information, which assists in the accurate classification and type casting of concepts.

The three concept levels ensure that a concept can be adorned with continuously more information.

Figure 4.2 shows business administrative, informatics, and industrial design concepts joined together. This illustrates that conceptually we can view chairs and sitting down in the same way as computers and information systems. This does not result in a simple figure, but it does show the similarities and differences.

Dragon1 model – Concept levels

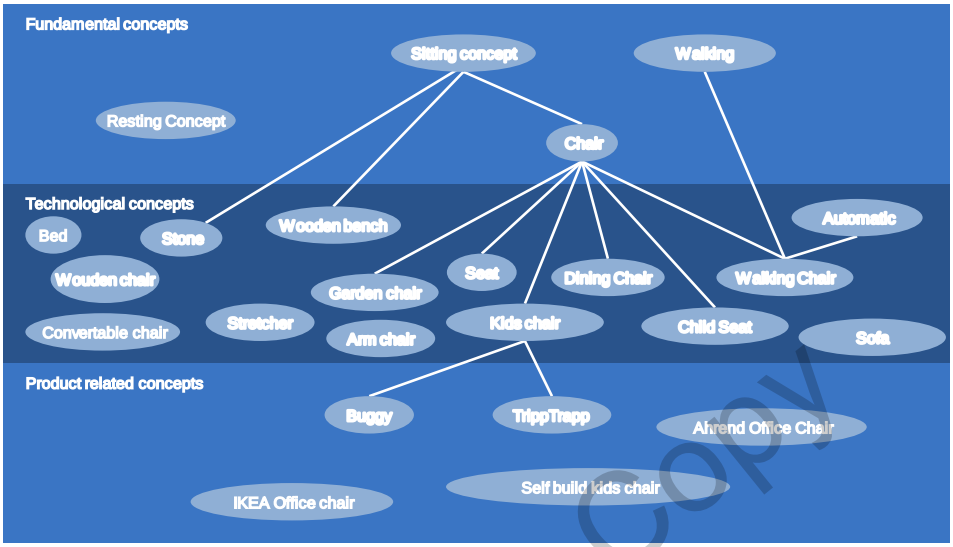

Figure 4.2. Example of concept layers.

Figure 4.2 shows an example of how the architect continuously details the concept 'chair' on a fundamental, a technological and a product related level. The concept 'chair' is presented at a fundamental level as – Fundamental concept chair. The concept 'chair' does not reflect anything yet about possible application, the technology or the supplier of the chair.

At a technological level we see different types of (technological) chairs, e.g. a child's chair, a sun chair and a garden chair. Because of available technical concepts the architect is able to make a choice of solutions without dependence on suppliers or supplier products. The result is that these types of concepts quickly become sustainable and future proof.

At supplier/product-dependent level we see the kind of chairs we can purchase in a store. Dragon1 recognizes the supplier/product-dependent concept level as the level where it can be made clear how enterprise architecture can be made independent of suppliers or products and whether this is necessary or not. The architect can at this stage, already legitimately include, which available and quality products should be obtained from certain suppliers, in the architecture design. However, the architect should realize that in this way he creates a dependency between client and supplier.

Note: The concept title sometimes displays the technology and sometimes a technology implicated by functionality. To the architect is it important to realize that each word used by him to indicate a concept name wittingly, or unwittingly, implicates functionality, performance and technology. The architect needs to be aware of this.

Why does the architect consider a 'chair' as a concept? The reason is because the architect wants to determine which type of chair is in conformance with the client's wishes or specifications. To address a chair as a concept, it becomes possible to design a new chair that meets certain performance and quality requirements. If existing chair concepts do not meet the requirements, the architect can combine, integrate and enhance existing concepts in order to arrive at a new chair concept, providing a model to create the ultimate chair required.

For example, by regarding a chair in a conceptual manner it is also possible to design a new chair that meets different requirements, e.g. a garden sun chair for children. Such a chair immediately becomes a small chair that has a reclining back, is weather-proof and has four legs and is made from teak or plastic.

The concept chair says nothing about the technology, but it appeals to an archetype of a seat with four legs. The client will ultimately decide which chair he chooses in accordance with his wishes and specifications. For instance, if a person needs to rest then this person requires something to sit on, a chair will be sufficient. But if a person wants to relax he wants to sit on something that looks like a comfortable, stable chair, such as an arm chair. If a person wants to relax in a reclining position, then that person prefers a sofa over a wooden chair.

To design an architecture of a structure embracing conflicting requirements, it is helpful to first design the structure's functionality and secondly to include the technical aspects. This is the main reason why an architect must be able to work with concepts that are still independent from technology.

An example of working with concepts becomes clear when an architect is faced with a request to design something that allows a client that flies and float. The architect will place the concepts flying and floating on the top fundamental level. At the technological level, he can then choose for a float-plane, a catamaran-helicopter or a hovercraft all in accordance with additional requirements pointing to one of the three technological concepts. Once the architect has chosen a technological concept, the architect can determine which suppliers and technical products best suit the client to construct the ultimate technological concept.

Quality, performance and component oriented concepts

Concepts are, as previously stated, ideas, abstractions, approaches or 'ways of working'. In Dragon1 a concept's core or a concept's entity is divided into four different types of concept: quality concepts, performance concepts, concepts of elements and concepts of components. There are more types of concepts but the ones mentioned here are actually very different and often reappear in the architect's practice.

A quality concept is about the way a certain quality can be realized, e.g. a safety concept. A safety concept is a concept that when applied, increases the safety of a structure. An architect can use different categories of quality aspects in order to determine which types of quality aspects are available to him or will be available in the future.

A performance concept or action concept covers the performance of a concept in a general or specific sense. An example of a performance concept is a purchasing concept, which states in general terms how to purchase something. A 24x7 sales concept implicates the way a certain performance guarantees 24x7 sales paradigm.

An elemental concept is a concept that describes, from a logical or functional point of view, how a certain operation can take place independent from any form of implementation, e.g. an insurance product concept.

A component concept is a concept that describes how a certain technical entity is created, operatively, constructively as well as decoratively, e.g. a digital insurance product concept. With a component oriented concept, certain logical functions are included next to certain technical constructs.

An example is a digital insurance product. In a component-oriented concept, there are certain logical functions fulfilled by technical constructs. A digital insurance product concept consists of a number of elements, which are integrated with components. Which components they are is described by the description or visualization of the digital insurance concept; also the description describes which advantages (quality and performance) the digital insurance concept delivers, which we get from the description and visualization of the concept.

In practice we see how an architect creates a total concept from a diversity of combined concepts, such as a flexible insurance concept, combined with a self service concept, a 24x7 sales concept, and a security concept. This total concept will then create the architecture of a solution.

From the dimensions, levels and types of concepts an architect is enabled to find his own way in reusing, determining the usefulness of existing concepts in order to arrive at new concepts for approval by the client.

The architect must, of course, understand how existing and proven concepts work before he can include them in his architecture design. If the architect incorporates new concepts, it is his responsibility to convey the functionality of the concept to the client and explain where the concept will be applied.

In itself a concept in fact does not represent implementation, but the source from which the concept arises does have an implementation. A concept is as such applicable to other and new situations.

Take an enterprise that sells insurance – the enterprise could be a national insurer or it could function as a local agent. National insurer and local agent are, in this instance, two concepts, which are independent from the actual implementation. But a person who knows how national insurers and local agents operate and how they cooperate together, knows what the enterprise could possibly look like when it becomes visible how one or the other concept performs; in other words, of which principles, elements and components the enterprise exists.

Dragon1 model – Concepts-and-principles meta model

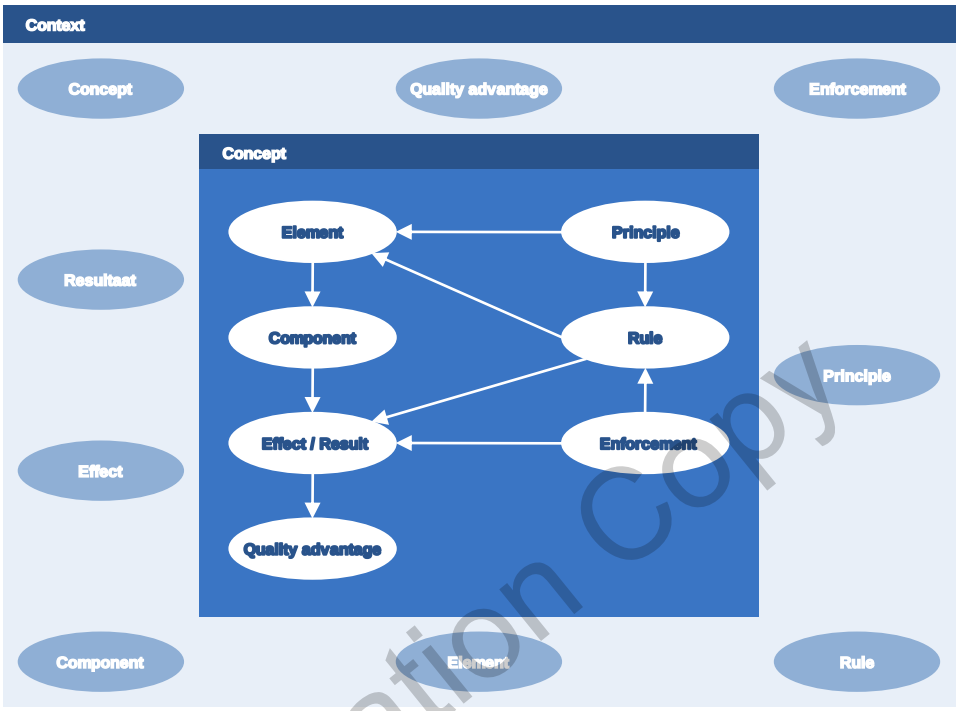

Figure 4.3. Concepts and principles.

In the **Concepts-and-principles meta model** entities such as element and component are shown as concept parts and as parts of an environment. The reason is because elements and components can exist freely in the setting of a concept or as a part of other concepts. For example, in a shed an old discarded table is used as a workbench, in the attic a similar table is used as a computer worktop. This shows that one single entity can, if used in a different context, have a different function or meaning.

By identifying a meta model as a model of concepts and principles it becomes clear how concepts and principles are inter-related. The performance of these concepts and principles and the ultimate choice of these principles are of strategic importance to the enterprise.

4.3.2. Fundamental concepts

By understanding fundamental concepts, the architect is able to explain the performance of concepts that appear and are maintained in nature. Concepts and

especially fundamental concepts are often archetype concepts. Archetype concepts are idealized primal models of solutions, which constitute the foundation of later variants. The archetype model of a house could be four walls with a roof, whereby windows and doors are optional. If something looks like that people will soon recognize it as a house and that they can live in it.

Examples of fundamental concepts are unique numbering, or sitting down or even the power of attraction. The law of individuality states that if something is numbered in a unique way it often becomes 'one-of-a-kind' and as such can be more identifiable. The old Romans began to allocate numbers to houses by writing a different number on each house. This way it was possible to control who had paid their taxes.

Nature, through physiology, compels people to accept that sitting down is a relaxing experience. The concept 'sitting down' consists of offering a surface for a human to rest or perform work in a relaxing environment. The word surface does not imply as yet another technology, product, supplier or implementation.

By using the concept of house numbers or a surface to sit on we do, however get a vague image of a solution, and also many degrees of freedom to determine or utilize technology. An architect likes to use fundamental concepts to remain totally free in determining solutions. The architect often links the necessity of choice for fundamental concepts in an architecture design directly to the needs, requirements or challenges from the client and other stakeholders.

With regards to identification by name, fundamental concepts can often be divided into quality-aspects oriented, element oriented, and action oriented. Added to this sub divisions can be made in terms of global concepts and detailed concepts.

- *A fundamental concept is a concept that offers, independent of new technology, a solution whereby maintaining the performance of a concept becomes a compelling aspect.*

4.3.3. Technological concepts

By understanding technological concepts the architect is able to explain the performance of concepts which emanate from certain technologies. Technological concepts are based on fundamental concepts. An architect will take stock of different technological concepts in order to complete a fundamental concept appropriately.

Energy saving bulbs and light bulbs, for instance, are technological concepts for the fundamental concept 'lighting'. An energy saving bulb is as a concept more durable than an ordinary bulb, but it is also more expensive as an investment. The needs, requirements, and challenges of the client and stakeholders will then determine the choice of bulb. Since the concept of the energy saving bulb is relatively new, we link it to certain technological solutions.

Chlorine free paper is also an example of technological concepts. Currently we don't consider this concept to be sustainable and future proof any more. That is why we don't often print emails on paper.

Technological concepts can also be divided into quality aspect oriented, element oriented and action oriented as well as into global and detailed categories.

- *A technological concept is a concept that is often inextricably connected to a new technology.*

4.3.4. Supplier/product-dependent concepts

By understanding product related concepts the architect is able to explain the operational aspects of the suppliers delivered products in which certain solutions are used. These concepts are based on technological concepts. An IKEA chair is a supplier/product related concept, as we associate it with self assembly items.

It is suggested not to use a product dependent or supplier dependent concept in architecture. It is up to the architect to decide if a concept is technological or not, or which alternative technological concepts are available.

An exception to the rule in almost every organization is with Microsoft. When we learn from business administration that an organization must not be dependent on suppliers we notice that as far as software products are concerned an exception is made for Microsoft. Microsoft has many innovative products in which standard concepts are used, which Microsoft has limited into 'non-open' Microsoft standard concepts.

Enterprises using a product in which a non-open standard concept is used may enjoy short term advantages, but from a business economic point of view it is not a sustainable or future proof investment proposition.

Product related concepts can be subdivided into quality aspect oriented, element oriented and action oriented as well as global and detailed categories.

- *A supplier/product-dependent concept is a concept that is linked to product or supplier of a product range. The use of a concept often causes unnecessary dependence on suppliers or the products.*

4.3.5. Total concept

The architect in his 'way of thinking' needs to reserve an important role for conceptualization, to consider problems from a conceptual level. When the architect has taken stock of sufficient concepts, and has made a selection of a possible solution, Dragon1 suggests seeing this as an action to create a total concept. By working with a total concept, the architect is able to communicate better with the client and explain the reasons why he choses to integrate certain specific concepts.

A total concept soon indicates the outlines of an architecture or the global architecture of an enterprise or a part of an enterprise. If a client looks at a total concept he will be able to visualize many aspects.

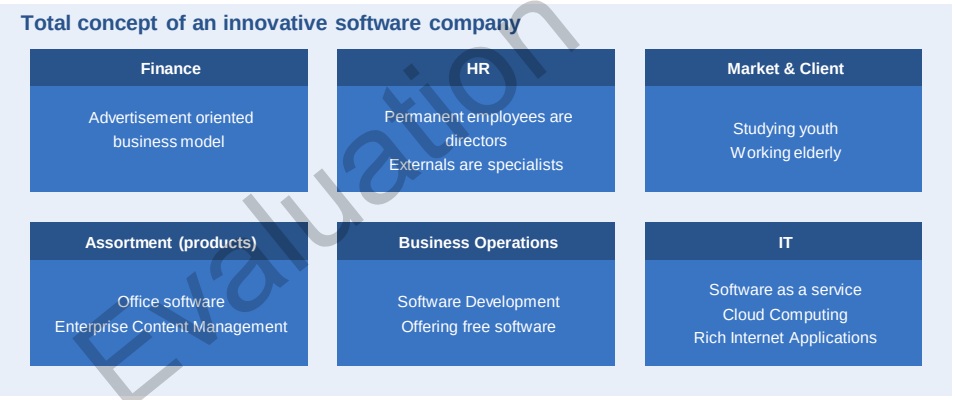

Figure 4.4. Example of a total concept for an innovative software company.

Figure 4.4 provides an example of how concepts fit together without causing a conflict. This total concept can now also be elaborated upon with concepts consisting of more details at technological level, but preferably not yet at a product level. The latter, only appears in the definitive architecture design.

The total concept shows in Figure 4.4 consists of two dimensions: 1) an enterprise domain model and, 2) allocation of technological concepts to a domain. Whether or not the enterprise or its products are attractive or perceived to be attractive, is not controlled by concepts. This is because the architect has not yet recognized

decorative concepts, such as a pay-off, information columns, or unity in the physical and digital world with a specific brand name.

In a total concept we notice hierarchy and decomposition of concepts. An architect must be aware that he continuously has to make implicit choices regarding decomposition and hierarchy. To do this in a more explicit way the total concept becomes less 'cut-and-paste' and 'copy-work' and more something of the architect can set his own seal upon.

In chapter five, we will elaborate further on the different ways of total concept model visualization. We will also cover the different perspectives of the total concept model. The architect needs a license to visualize certain versions and parts of the total concept in order to properly 'sell' to or to communicate with, the client, steering committee and other stakeholders.

- *A total concept is the consistent total of collaborating concepts, which the architect uses as a basis for the architecture design. In fact, we could soon see hundreds of these concepts or more, as well as large parts or components of the architecture.*

Architects realize that technology alone is not sufficient. A technical solution must also be accompanied by enterprise restructuring. An architect will, therefore, in a total concept not only focus on IT solutions, but also focus on business solutions such as improving competence and retraining employees.

Dragon1 model – Concepts framework

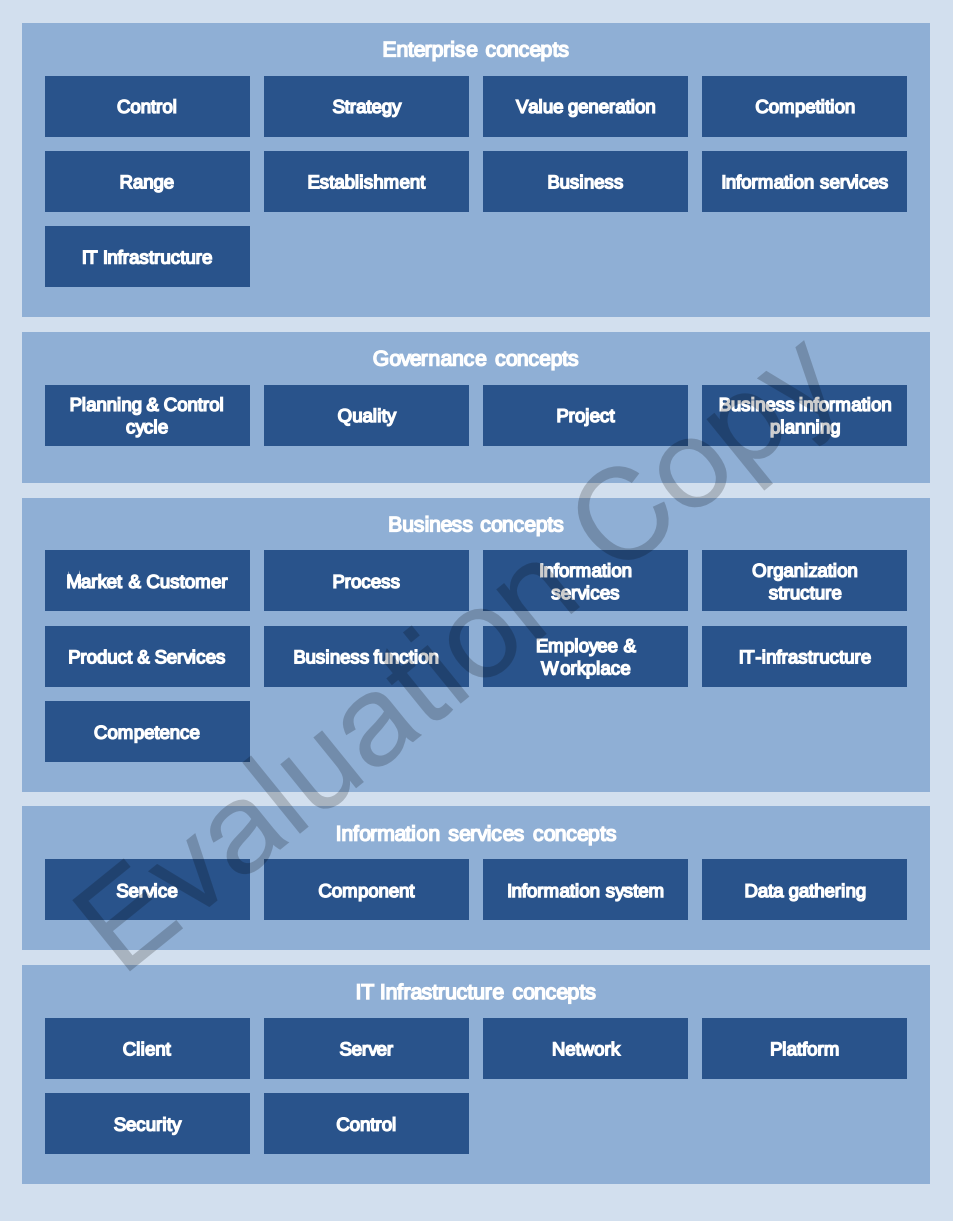

Figure 4.5. Concepts framework.

The **Concepts framework model** shows an overview of common enterprise architecture concepts. Concepts could also be referred to as independent building

blocks for generic implementation. When communicating with stakeholders it is sometimes advisable to change the abstract term 'concept' in this way. Moreover, all the concepts in this model can be described in a more abstract way, or in more detail. It depends basically on what the architect recognizes given a certain situation, or considers necessary and where the client recognizes his personal ideas and its requirements.

Figure 4.5 shows an example of a server which is a concept that is utilized within an IT structure. Say, that there are only application servers within an enterprise and no file server or print server; the architect should then have made the concept 'server' clearer by denoting it as 'file server.' Next to server the client is also recognized as a concept. Server and client are both specializations of the concept 'computer'. The architect, however, has decided not to choose the 'computer' concept but to recognize the two specializations 'client' and 'server' as concepts in the architecture of the IT infrastructure.

The architect operates from a certain benchmark when making choices in specialization or generalization of concepts: e.g., what does the enterprise want to become independent from, or what must the enterprise become independent from? Clearly, the choices of the architect are of great impact here as indeed they allow room for strategy and this will speed up, or unwittingly delay or block progress.

Therefore, architects use reference architectures to assess and test how globally or detailed, and how generically or specific they need to recognize concepts in an architecture. In Figure 4.5 the framework has been built from different levels in the architecture. In one architecture level we find concepts which are inter-connected. The information architecture level, which includes information facility concepts, as well as the technical architecture level with its IT infrastructure concepts, is a level from which the architect takes conceptually into consideration the information facility and IT infrastructure elements of the enterprise structure.

Taking the above into consideration, working in terms of architecture levels it becomes possible to assess a structure or total complex, such as an enterprise. As such it is as if the architect is wearing blinkers to filter out certain parts or aspects that are not important for the moment, so as not to cause confusion.

Both business administration and informatics are full of concepts. The more knowledge the architect has of these concepts, the more the architect can evaluate whether an enterprise works with or without these concepts. The architect performs the evaluation on the basis of asking himself questions, to formulate hypotheses and

test them on their feasibility in the enterprise. An example of a hypothesis is: All work performed in the enterprise takes place conform recognized and identified activities. With the help of this hypothesis it can be investigated whether the business process concept is used everywhere in the enterprise.

4.4 Principles

In Dragon1 the recognition and identification of principles is considered important for the architect. By being able to see associated principles that are hidden behind the concepts, the architect can make the appropriate choice of the concepts he needs. This paragraph covers first what is meant by principles and which types are recognized by Dragon1, and following this, the specific parts of principles is taken into account.

4.4.1 Overview of principle types

It is the task of an architect to deliver benefits resulting from working with concepts, in the right context and through the right implementation; it should apply in the structure that they design. The architect must be very clear about the functionality, the consequences or the impact of the result, that are produced by concepts. It is also important that the architect knows the 'what' that makes a good concept, the enforcement mechanism and the elements, the components, the objects and the rules that must be minimally realized. Ultimately the components and objects are implemented with technical products

A principle describes the enforced operation of the entire concept or from a part of the concept. A principle describes a 'cause and effect' relation (by ... [this]... then ... [that]... happens). Dragon1 calls a principle that describes the entire operation of a concept the 'first principle.' An architect needs to find the 'first principle' because it allows the essence of the concept to be quickly understood and conveyed.

An example of a first principle is the loose-coupling principle of services: **by** offering services without the consumer of the service becoming dependent on the way by which the service is delivered by the service provider, the service provider **ensures that** the process of service provision can be changed often and better with minimal impact for the consumer **by which** the service of the enterprise becomes more adaptive as a whole.

Say an enterprise needs a webshop for her products and that this webshop must be coupled to the sales system on the basis of the web services. The way by which a supplier creates a webshop in accordance with a certain technology is less important than the fact that all communication takes place on the basis of web services, with unbounded loose couplings between systems. Consumers of a webshop are never

really interested how and with which technology a webshop is constructed. What they want to know is which types of standard interfaces or protocols are used for communicating with the webshop. If the supplier of the webshop, for various reasons, changes over to a new website platform or to another version of the webshop, this can now have only a minimal impact on the buyer of the webshop. This will be the case as long as the loose-coupling principle by way of communication between systems remains operational.

Sometimes a concept is so elaborate that the concept consists of part concepts that in turn are also independent concepts in their own right. This means that principles are applicable to the part concepts as well.

The architect usually looks at four types of principles to maintain control over the quality of the structure that he designs. This concerns the concept principles, the system design principles and the principles of the yet to be designed systems. Also, the architect looks at the principles in other areas to learn and to identify candidate principles for reuse.

Dragon1 recognizes different types of principles in order for the architect to be able to apply a certain type of principle in a most efficient and purposeful way. By principles it is important to realize that everything revolves around whatever principles already exist and has been determined, in other words, which result belongs to, or is produced by the principle.

It must be recognized that a client sometimes wishes or even demands (in terms of requirements) that a certain principle is taken or is to be taken into account. Requirements will then be labeled as principles. But requirements appear in the table of requirements and principles are part of the architecture. In the table of requirements, certain principles can be demanded. It is essential to keep them well apart.

If it is not known yet which consequences a principle has or produces but the principle is already formulated, then it is often a hypothesis or axiom. A hypothesis is a claim that still needs to be proven; an axiom is a generally accepted hypothesis that can be negated at any moment. A (scientific) principle is to all intents and purposes difficult to negate for hundreds of years.

It is also possible that rules, the agreement between two or more parties, are labeled as principles. An agreement, however, does not say anything about the degree to which it is maintained, as opposed to a principle. An agreement does not dictate

either the degree to which a person can expect that everybody will keep to the agreement. Contrary to the aforementioned, in law however, this is the case that one may rely that an agreement is maintained. With rules such as regulations and directives, this is not the case.

An example of a rule, which is labeled as a principle and which is used as a principle, is 'a single request for information': 'We only request our clients to provide certain information only once; and we administrate these (details) in such a way that we can always reuse them later without having to send another request'. This looks like a good principle, only it isn't a principle it is an assumption, rule or requirement. There is an agreement over how this always should transpire. However it appears from this statement how the agreement will be enforced. In order for it to be always valid. It also appears from this statement what the frequent reuse of this information is. Why would this enterprise invest in developing a system that achieves all this?

When an architect takes a good look at this rule, he can discover the intended principles that are meant, namely the principles 'lower cost through reusing information' and the principle 'efficiency by reusing information'. By administrating the information only once and then reusing the same information, it ensures that the enterprise saves time by avoiding repeated administration of information and by making fewer mistakes by which holding incorrect versions of the information will be less likely.

Figure 4.6 shows many examples of the types of principles present in the entities of an enterprise.

Dragon1 model – Types of principles

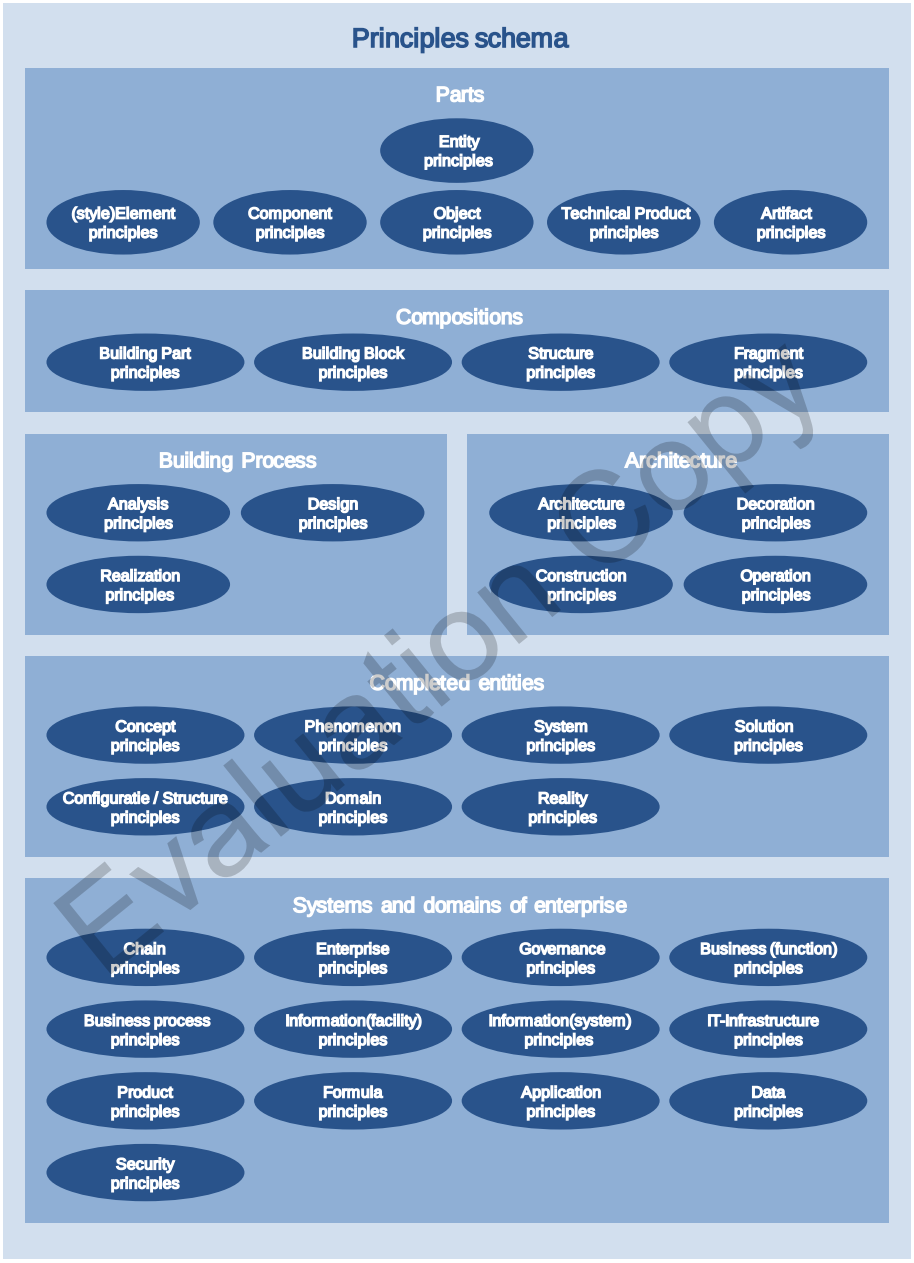

Figure 4.6. Types of principles.

The model **Types of principles** shows an example of how different types of principles can be classified. There are more principles shown in the model than mentioned in this chapter. The 'why' and function of these types of principles can be simply derived from the principles covered in this chapter, as follows.

4.4.2 Concept principle

A concept principle is the explanation or description of the enforced operation of a concept creating a certain result. A concept principle is also called the operation principle of a concept.

The principle 'modularity' is an example of a principle of the constructive technological concept 'modulair construction'. Customers increasingly require their suppliers provide solutions that offer maximum flexibility. The principle of modularity can be described as follows: 'by always producing solutions from an environment of robust quality control pertaining to individual specialized and optional components, which can be assembled quickly into various combinations, it can be assured that solutions will emanate from different people at different locations. This allows the supplier to meet customer demand for maximum flexibility. This is the reason why an architect will often choose for a modular solution when a customer demands flexibility.

We note as well, how a principle can be formulated with the assistance of certain sentence-constructions to describe the maintained operability of a concept. With the assistance of this formulation, we also note which elements and components make up a concept and which rules apply.

The concept must explicitly mention the way by which the operation is enforced, as only then will it become clear when the principle is applicable and in which context. The success of an enterprise depends on the degree to which a concept is enforced and that it is adequately implemented (within the enterprise).

- *A concept principle is a principle that shows the way a concept has been put together, how it works or what has materialized in the concept, from which a certain result is always produced or a certain effect is noticeable.*

4.4.3 Design principle

The architect creates architecture designs of structures. In order to realize a structure with a high level of quality the architect utilizes certain principles in his designs. Dragon1 calls these principles 'design principles'.

An example of an enterprise design principle is: 'the principle of independence by way of service procurement'. The design of the operating model must include the procurement of services instead of highly prioritized technology, solutions or products. This way the enterprise remains independent of the underlying technology, solutions or products that are inter-related to the procurement services.

An example of an information design principle is: 'to incorporate a hosting provider's groupware solution which links internal information systems via a generic XML interface. First the supplier incorporates GroupWise and then changes over to MS-SharePoint. To the customer this means a different user interface and increased functionality. The supplier has to transfer data and structures, whilst keeping to the XML interface definition. The customer, himself, does not require knowledge of the underlying technology of the groupware service that has been procured. This is a prime example of sustainable procurement, while reducing dependence on technology.

An architect must choose design principles that realize or meet quality requirements, strategic starting points and pre-conditons of the envisaged structure. A constraint is that all matters pertaining to the client's/owner's mission, vision, identity, culture and preferences are appropriately translated in accordance with their requirements – principles in phenomena, concepts and existing systems are models of design principles.

All acceptable architectures consist of a myriad of design principles. The architects should limit, as much as possible, inventing their own creative design principles, and as much as possible incorporate best practice design principles. This will provide knowledge, assurance and predictability during the application of design principles. They have already been tried and tested.

Dragon1 recommends identification of architecture design principles in each accepted architecture in an architecture framework. This applies to all types of principles.

For the purpose of readability, we reiterate the definition of a design principle.

- *A design principle is a principle that is applicable to the environment of a structure and as such influences the structure. The architect must take these principles consciously into account when designing and realizing a structure.*

4.4.4 Architecture principle

By recognizing architecture principles, the architect can strip concept principles and design principles from one structure, or from a certain discipline of its context, in order to reuse the principles in another structure or as part of another discipline. As such, an architect is capable of coupling different architecture principles to each ambition and each strategic starting point of an enterprise. This way the architect ensures that strategy constitutes the basis for architecture in all the nooks and crannies of the enterprise.

An example of an architecture principle is: 'information leads to knowledge, and knowledge leads to power.' This principle is applicable everywhere. This applies to people, to software systems and enterprises. The principle dictates that if an enterprise possesses more information than other enterprises about a certain subject then the enterprise is able to take better decisions regarding this subject. And if someone is able to make better decisions about a subject, the more someone will be given higher responsibilities. Architects use architecture principles mainly to group design principles and to cluster them at a higher level of abstraction.

- *An architecture principle is a concept or design principle that is devoid of all contexts or a design principle that can be used always and everywhere in a particular system. An architecture principle is an universal principle that is applied in the holistic enterprise and therefore a comprehensive statement about the 'construction' of the enterprise.*

4.4.5 Reality principle

Reality principles state what is valid in a system that has yet to be designed or realized in a system. With a reality principle the enforcement must be stated explicitly to make the principle valid. This is contrary to design principles. Design principles dictate what is valid in the context of a yet to be designed system. With design principles the enforcement is already maintained in the concext by the yet to be designed system.

The reality principles state how the enterprise concepts behave. Herewith, the architect is enabled to detect fundamental inter-relational flaws and solve them forever. An architect is often faced by a mish-mash of reality principles, which maintain certain undesired situations. A clear view of this makes the finding of solutions to design issues much easier.

Dragon1 differentiates between AS-IS and TO-BE reality principles. AS-IS reality principles, desired or undesired, are principles which already are present in an existing system. On the other hand, TO-BE reality principles are principles that will be applied in a system.

TO-BE reality principles are different from design principles as a design principle covers an operational mechanism and because it is also applicable outside the structure that will be designed. A reality principle only covers the operational mechanism in a yet to be designed structure.

An example of an undesired 'AS-IS' principle regarding enterprise activity is: 'because our products are not supplied with accompanying digital information formats, additional costs will be incurred for printing information media resulting in lower profit margins'. Naturally the enterprise wants to improve on this principle.

An example of a design principle regarding enterprise operations is: by supplying our products with digital information it is ensured that the costs at the point of sales are lower resulting in higher profit margins. This principle exists outside the enterprise, even before they begin with the design of an architecture.

An example of a 'TO-BE' business reality principle of an enterprise is: by always providing digital information about our products and providing them via the internet our business enjoys lower sales costs resulting in higher profit margins. This business reality principle only applies with the passage of time in the business.

It pays for an architect to know which reality principles are at play in all the enterprise's systems and in fact all the elements of the enterprise configuration model. This belongs to the 'AS-IS' architecture and the enterprise architecture design.

All being equal, the 'TO-BE' reality principles of a system Z that is yet to be realized are parallel on a 1-to-1 basis with the 'AS-IS' reality principles of the TO-BE realized system Z. However, in reality it seems that this is not often the case. Because 'TO-BE' reality principles are sometimes not used properly or a system has not been realized properly, which can lead to wrong and/or new principles being applied to

the structure next to intended reality principles. One more reason to take stock of the principles that appear in reality in every structure and to match them with the principles used in the design of the architecture.

It is important for an architect to know what the current situation is in an enterprise regarding principles, especially if an architect wants to assist the enterprise in getting rid of persistent, wrong principles. Therefore, the architect must know exactly which principles are valid, as well as knowing what and who enforces these principles, and to understand which elements they consist of and why the effect or result of these principles do not contribute in realizing a strategy.

For all architectures in an architecture framework, the architect must have at his disposal a top ten 'AS-IS' reality principles, design principles as well as 'TO-BE' reality principles. With every architecture design that the architect makes he will require these principles as a framework.

- *A reality principle is a principle that states what the enforced way of working is within a realized system (or part of that system). In this case, a reality principle denotes a principle that applies to the current situation in an enterprise.*

4.4.6 Perception of principles

In this paragraph we describe how the architect can perceive principles, concepts and certain phenomena as well as identify them in different types of systems. The following systems are considered: nature, enterprises and structures.

In order to create an architecture design, it is important to identify principles. It requires knowledge, experience and abilities to recognize principles and to study them. Once an architect identifies a principle it becomes easier to utilize principles in an architecture design or as criteria for the selection of a concept.

Dragon1 model – Perception of principles

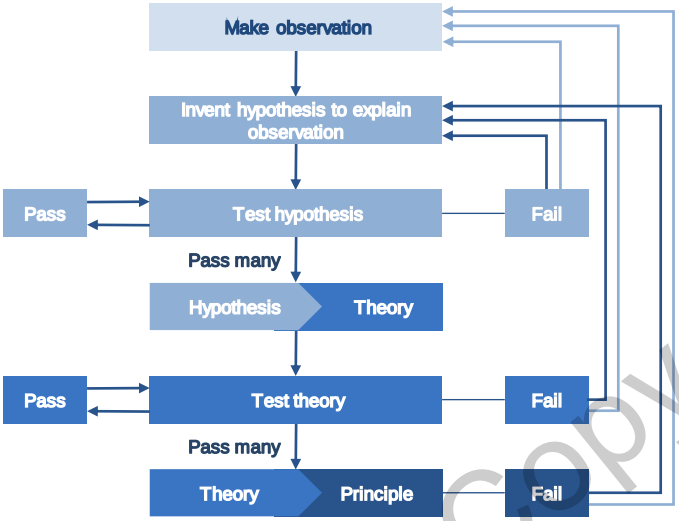

Figure 4.7. Perception of principles.

The **Perception of principles model** shows an example of how, by way of testing and honing statements, an architect is able to arrive at a correctly formulated principle. The architect's knowledge, experience and ability regarding business administration and information technology determine which principle, (operational) working of systems, concepts and phenomena he perceives as being important and which questions he needs to pose regarding the principles.

Regarding the perception of principles, Dragon 1 takes into account the context of these principles. The context may vary and consist of all types of situations, environments and locations.

Situations

The architect includes in his architecture design the various situations that exist within an enterprise.
An example of a working situation is: the daily work routine, consisting of: arriving at the workplace, logging in and starting up an application.
An example of a user situation is: sitting in the canteen at lunch, requesting a building permit on the internet via a PDA, to build for structure extension to his home.

The architect must also differentiate between the physical and digital user situations and work situations.

An example of a user situation: an user session for ordering a book through internet. An example of a work situation: during the approval of an application for a social security payment.

Every situation is running in a timeframe from (an instant) moment to a period (longer time spam). A period takes place over a relatively long period of time. A moment takes place in a relatively short time.

- *A user situation is a moment in time whereby the user of products, services and goods plays a prominent role. It involves a certain relationship, coherence and association with other actors in the context of an environment.*

The architect must differentiate between situations, environments and sites, because customers and workers use different resources, facilities and infrastructure at different moments and in different ways. These activities pose indirect demands, while people pose direct demands in a work situation. For instance, at the employee's place of work different rules apply than in a user situation at home via the internet. At an office location or at a home location Different rules apply. In an office or in a factory, different rules apply than in an user environment on the street.

- *A work situation is the user situation of an employee. A user situation is a situation of the customer.*

The structures designed by an architect, such as businesses, information facilities and IT infrastructures are entities that are used, in different situations, environments and locations. During the design, the architect must be well aware of these situations, environments and locations, in order to analyze them as a context in the architecture design.

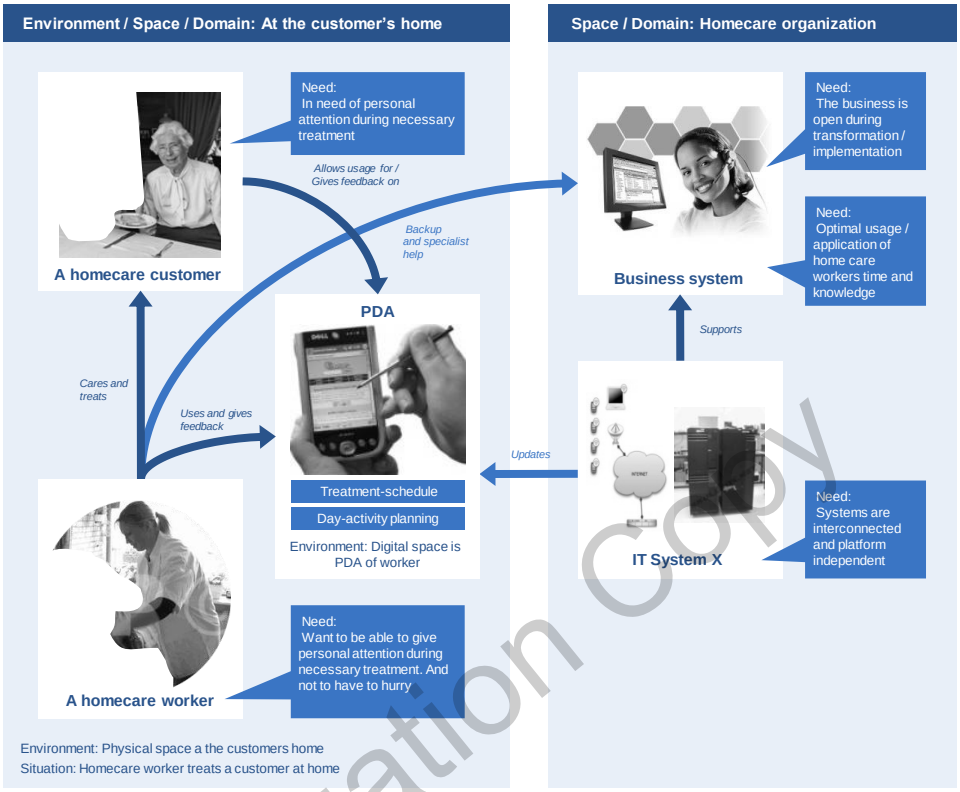

Figure 4.8. Example of principles and concepts in a Care practice.

Figure 4.8 shows an example of different situations, environments and locations. It depicts a situation in which a solution of chain reversal and the personal planning of activities by another person (employee) are made possible with a PDA (Personal Digital Assistant – which is basically an all comprehensive mobile telephone). For instance, in the case of a care provider, at a patient's home, at a patient's home a care provider can use a PDA to work more efficiently. In order to make the solutions work more efficiently the architect needs elements and components important to the concept of visualizing reverse care activity planning indicative to an in situ situation.

In Figure 4.8 it becomes clear that the PDA not only uses a network and database server to obtain planning and activities, it also allows the care provider to revert back to a manned support center. It then soon becomes obvious which demands the PDA has to comply with regarding availability, security, robustness, convenience and comfort of use.

An architect should, at all times, take three different situations into account: an average situation, an exceptional situation and an extreme situation. He needs to include these situations into his design. At different moments and periods during a day, week, month or year, normal and extreme situations will take place. A system must be designed accordingly and in such a way that it remains in tune with such situations and that it provides desired and predictable services at all times.

User space and work space

Situations always take place in certain spaces. These can be physical or digital spaces. Physical spaces at work are for instance the counter, the department or the back office. A digital space is for example, the social network, the intranet, the groupware environment or business system.

- *A user space is a protected environment in which a user has access to certain resources, facilities and infrastructure. A user space is primarily meant for customers. A workspace is a user space meant for employees.*

Locations (Environment / Sites)

All structures and structural parts that are designed by an architect exist at one or more geographical locations. Even if currently everything can be done digitally or virtually, there still are many architecturally designed solutions that are not 100% independent of place, time, space and situation. Therefore the architect has to include these dimensions of a structure into his design.

- *A location is a geographical place where there are user spaces and/or work spaces, and where user situation and work situations take place with customers and employees.*

An architect designs situations, locations and spaces that are created by him primarily with resources infrastructures and facilities. As such various situations could take place in these spaces and locations by way of, for instance, predefined scenarios.

Dragon1 model – Location-space-situation

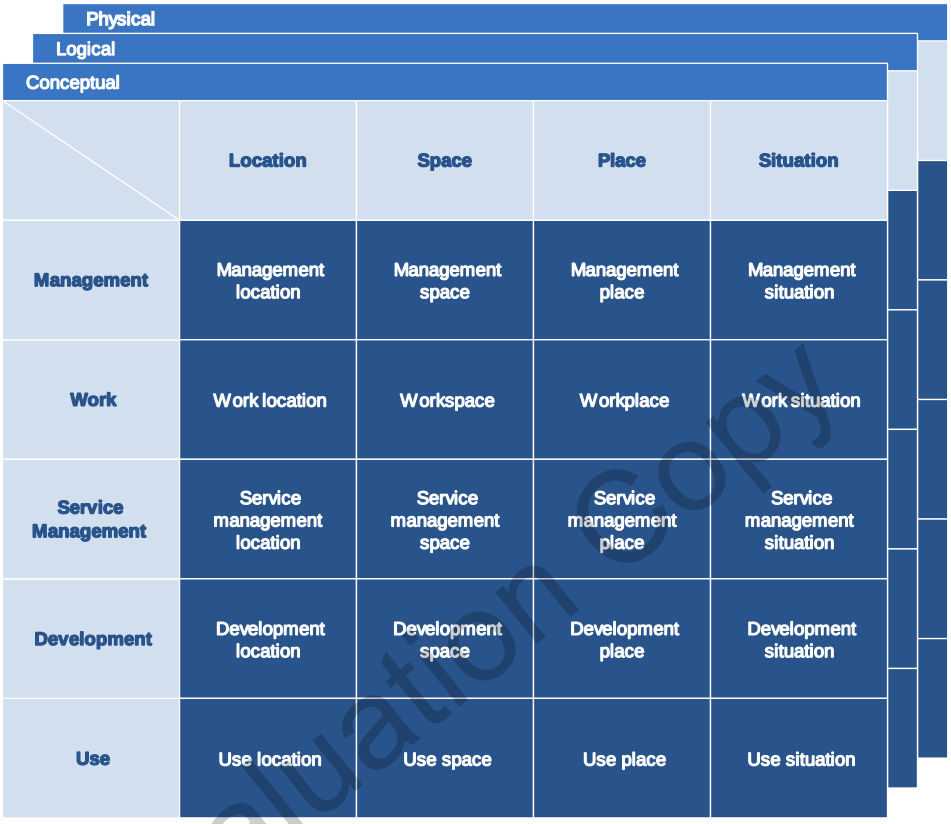

Figure 4.9. Examples of different types of locations, spaces and situations.

The model **Location-space-situation** incorporates a diagram that shows which types of spaces, locations and situations an architect can identify from a generic point of view. The more the architect recognizes these dimensions, the better the architecture design becomes. Say that an information system is used by customers, managers, administrators, developers and operational employees. Then it is important to identify types of actors for these target groups. The actors determine in turn at which spaces and at what location the specific demands made regarding the use of information systems.

4.4.7 Enforcement Mechanism

As mentioned before a principle is an enforced working of a concept, structure or phenomenon. Because a principle is an enforced activity of a concept, structure or phenomenon the architect must, when describing the principle explicitly mention the part that is responsible for enforcement. The more an architect is aware of the element which takes care of the enforced activity the better he is able to implement principles.

- *An enforcement mechanism is the total sum of the internal and external elements that take care of the consistent way of working of a concept within a given context.*

There are different types of enforcement, for instance the builtin enforcement with people such as ethics, decency and the value of civilized behavior. In nature the powers of nature take care of enforcement. As for systems developed by humans, certain constructions and functions take care of enforcement.

Because the enforcement of a principle could fail, Dragon1 discusses the probability that a principle works in a certain way. The probability factor that a principle works in a certain way is greater than 0.7 (>0.7)

Architects must be aware that enforcement of a principle is the responsibility of people and that as such it is very possible that a principle does not always work in the same way.

Everybody faces different enforcement mechanisms on a daily basis. For instance, human conscience and discipline enables humans to conform to culturally, socially and legally bound rules. The conscience and discipline as such often compel somebody to react in a certain way, however conscience and discipline are not infallible.

Where conscience and discipline fail the authorities use different forms of enforcement, for instance police with cameras to detect and punish law violations, to ensure the way in which something is required to function or needs to take place is only temporarily disturbed.

Enterprises have their own ways of applying supervision and enforcement, which are necessary to ensure that certain principles are applied and maintained. In this case we think in terms of the security of a structure where only people with a pass are

allowed access inside the structure; or, agreeing to a project budget for an information system only when an approved business case has been published.

4.4.8 Phenomena

Dragon1 also recognizes phenomena next to concepts and structures, because the architect needs to into take account that they will appear sooner or later. A phenomenon is a fact or situation that is observed to exist or happen. Phenomena not only apply to situations such as a storm, rain or electricity interruption but also to customer loyalty, 'dilapidation' in terms of the deterioration of a system that has not been managed or maintained properly and it could also apply to 'an increase in complexity' – caused when the mutual dependencies between elements of a system are unnecessarily high.

An architect sometimes has to defy the laws of nature: he has to make an architecture design of a structure that is safe and open, flexible and stable. The architect, therefore, must also understand and control principles in a phenomenon, which will interact with a yet to be designed structure. The architect must also take into consideration exceptional situations in which certain phenomena may exist.

For instance, we look at a computer network. It may occure that employees, for whatever reason, demand the same file at the same time. An architect will normally design a network on the basis of three situations: common use, little use or heavy use. However, the next two situations are usually not taken into consideration in the design of a network 'extreme excessive use' and 'never-used'. When a situation presents itself when extreme usage occurs then there is a great problem of waiting times and file-locking conflicts.

With phenomena we can predict events and behavior. That is why an architect must be aware of the phenomena which appear in the vicinity of a structure and within the structure itself. An example of a phenomenon is chaos. If one does not demand data quality within a system then this will cause the database to become unfit for purpose within a very short time.

The architect has to use and must have knowledge of phenomena and protect the structure accordingly. If only we were to look at an enterprise from a phenomena point of view it will produce some interesting insights.

4.4.9 Rationales

In a structure we can find different principles in force. These principles work for a particular reason. The underlying reasons for the way these principles work in a particular way are defined by the Dragon1 method as a 'Dragon1 rationale'.

In a rationale a context often reappears. Outside this context a principle hardly ever works. The enforcement mechanism is not effective, in this instance. By looking at, for instance of a reality principle it becomes clear why certain design principles and concept principles do no work in practice within an enterprise.

An example of a rationale is: the success rate of an IT project will increase only if there is strict adherence to receiving approval for the business case of a project start up. However, if there is no or hardly any knowledge of writing a business case, then it will be difficult to set a business case as a precondition for a project start up. It is the architect's task and responsibility to recognize where there is lack of knowledge in an enterprise as well as the availability of knowledge.

- *A rationale is the basis that underpins the way a principle works the way it does.*

4.4.10 Rules

An architect uses rules such as business rules and IT rules. A rule is an agreement between two or more parties regarding a relationship between entities, whereby failure to comply with the agreement will invoke certain sanctions. This way a direction is a rule without sanction, a regulation is a rule with an average and a law is a rule with a heavy sanction.

Design principles consist of rules. These rules consist, for instance, of business and design rules in order to make a principle implementable.

For instance, take the following principle: 'a high level of agility due to modularity'. This will be translated by the architect into a business rule; 'an enterprise process may never directly utilize information resources from other business processes but must address its own resources or generic information services'. When making an architecture design the architect may use the design rule: 'every business process possesses its own information resources and makes use of common information resources'. If an architect works in this way, he will enable the enterprise to change

over to another business process which entails the support of different information systems, without the other business processes being adversely affected.

Another principle is for instance: 'controlled facilities assure quality of operations'. In many enterprises there are departments who will claim to experience difficulties with standard information services or with specific information services for their business functions. We often see this after a merger has taken place. A department may then decide wittingly or unwittingly to engage in a stance against information policy and work in support of an application architecture that is not permitted and to create and operate a new information system to manage the new information provision and even to link to existing enterprise-wide information services. We then see everyone using their own rules which result in thwarting the official principle.

An architect can now provide decision makers in an enterprise with evidence proving that an unmanaged specific information provision negatively affects the quality of the information that is meant for customers and top executives. Each information system should, as a matter of fact, at some point be allowed to develop or be adapted. Non managed or illegal information systems will possibly have an inferior quality and a non-standard technology, which will incur high cost when adapted. It then becomes important for the decision maker to make a quick decision to place the information system in a controlled environment on the basis of the architect's visualization. And, thus will comply with the approved application architecture, and as such make it clear which rules must be adapted.

A decomposition and a hierarchy of rules can be made in an enterprise. In reality there are so many written and unwritten rules that there always will be rules that will cause a conflict in the enterprise. The architect appreciates the challenge to create a structure in conformance with the rules, and to assure that the structure incorporates rules, which comply with the rules of the enterprise.

As with principles and concepts rules can be global, detailed, specific and generic. Rules can be integral, but also bound to systems and domains.

Dragon1 recognizes from a generic point of view different types of rules for common architecture domains, systems and elements. Hereby we think of:

- **Business rules** within which are included product rules, process rules, organizational rules and service rules.
- **Information rules** within which are included, data rules, application rules, and interface rules.

- **IT infrastructure rules,** within which are included, network rules, security rules, platform rules, and environmental rules.

To make rules pliable, enterprises choose for open and international standards. They contain large groups of rules that make it clear to everyone what to abide by. If the architect is aware that the enterprise has installed the process "configuration management" for the technical control of the IT infrastructure, he can take it for granted that an up to date configuration database is in operation in the enterprise. This is a database where all elements, components, objects and technical products of the enterprise are inter-related.

The architect will make clear from his architecture design which design rules count, which design decisions have been made and which design criteria have been used. The architect will also make clear which business rules, information rules and IT infrastructure rules will count in the structure.

4.5 Enterprises as structures

An enterprise consisting of one or more businesses, that is responsible for the execution of its activities, is a conglomerate of many activities, such as processes, products, services, people, systems and infrastructure.

4.5.1. An integral approach of an enterprise as structure

Figure 4.10 provides an exploded view of an enterprise which consists of many entities.

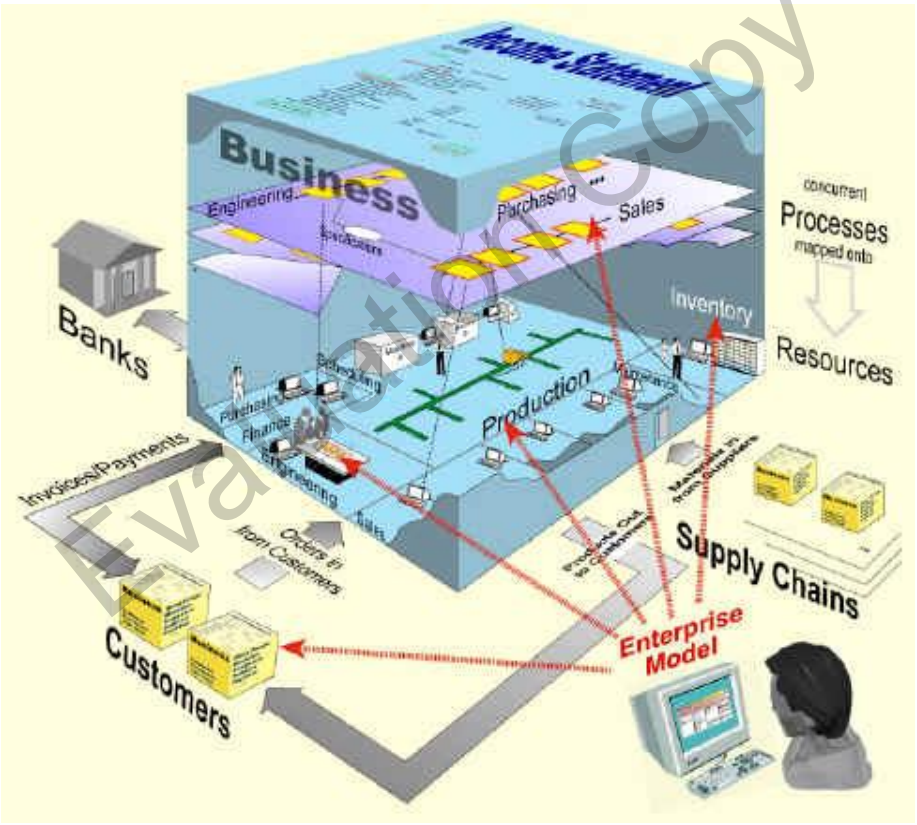

Figure 4.10. Enterprise as structural model and integral system.

When an architect designs a structure to consist as a part of an enterprise as shown in Figure 4.10, he must take into account the entities presented in this example. The solution designed by the architect to provision a set of requirements and enterprise

objectives normally consists of a business section with processes, employees, services and products, as well as an IT section with information systems, networks, and computers.

Dragon1 prominently recognizes integral business and IT solutions, as increasingly enterprise solutions consist of combinations of business and IT entities.

A mistake that was often observed in the '80s and '90s was that certain IT-projects failed to pay attention to the effort needed for business entities to change along with the other changes in technology. In some cases this resulted in failed introductions of entire computer systems that could not be operated, or remained out of commission.

- *An integral business or IT solution is a solution consisting of business and IT entities.*

Architects create models of the enterprise, the environment and the structure. All three models are then combined into a harmonious entity. The model Coherence in Architecture explains clearly how. Within the confines of large enterprises the environment of a structure often requires various levels.

Dragon1 model – Coherence in Architecture

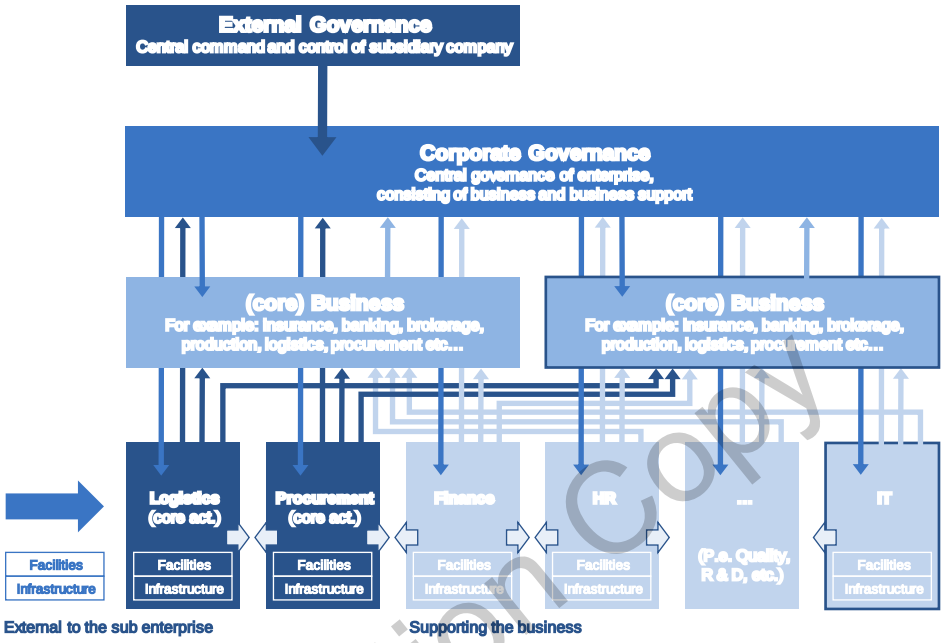

Figure 4.11. Coherence in architecture.

The **Coherence in architecture model** includes an example of cooperation between environment, the enterprise and the structure. We perceive an enterprise that is internally and centrally controlled. When an architect has to design a human resources or financial information system, he has to take into account vision, strategy, architecture, rules and policy, not only pertaining to the business in question but also regarding the internal and external control of the enterprise. If the architect does not take this into account at all, or insufficiently, the introduction and utilization of the system will cause problems, or even cause the system from being commissioned.

Or, if an IT department has certain rules concerning the procurement of solutions, it is a given fact that if this is policy, the procurement department has to act accordingly. If a procurement department requires new IT support and it is subject to policy, IT rules will apply. It proved as such that everything in an enterprise is interdependent and follows a so-called contingency theory.

The architect needs to employ a wider perspective than merely business activities and IT if he is to create architecture designs that are realizable.

4.5.2. Structure

An architect prepares to create an architecture design of a structure. Within an enterprise a structure often consists of a combination of business and IT entities, such as information systems.

Integral solutions such as CRM consist of renewed business processes, such as a sales process and information systems pertaining to supplier contact, history, and 'Name Address City' information administration.

These integral solutions form a coherent set of entities, and as such it is important to view them as structures that can and will be used for some time, and even expanded, replaced or renewed without causing a disturbance.

By regarding an integral solution to be like a structure and as such, consisting of various building components, architects, independently from each other, can perceive the structure's construction design, operation aspects as well as decorative aspects, or the individual building components belonging together. This also enables architects to design each aspect independently.

In case this is still too complex, the structure can be divided into building blocks, or its specific problems can be treated as fragments and mapped as such. This way problems can then be individually solved.

Dragon1 model - Structure

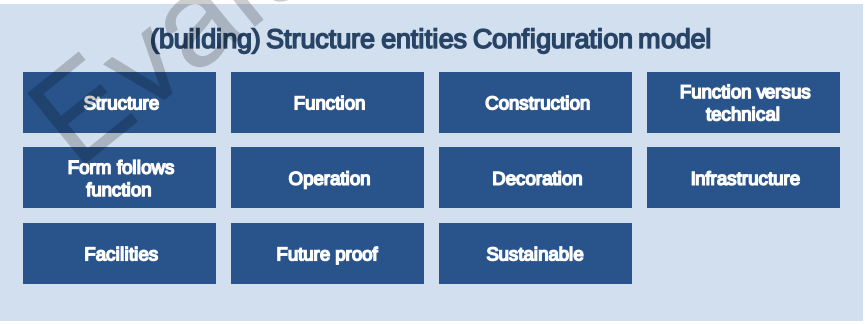

Figure 4.12. Structure.

The **Structure model** includes an example of which entities constitute a structure. Some structure entities constitute construction, other entities constitute operations and some other constitute decoration. This distinction explains a structure's three functional types.

Examples of a structure's constructive entities are: components which provide a framework for strength, resilience and division in an enterprise, and they provide a platform for agreements, business rules, strategic starting points, system perimeters, domain perimeters, policy frameworks, supervision and enforcement.

- *A structure's construction is a total of entities that resist certain forces or impacts to a structure. The stronger the construction, the better its strength and resilience. The structure's construction enables a structure's constructive functionality.*

Operative entities are often considered facilities in a structure. They represent a certain meaningful utility function to a structure's users. Constructive entities do not possess this feature, or possess it in an indirect way or hardly at all. Constructive entities are of little or no operative importance to customers and employees regarding the operational use of products and services. Operative entities inextricably connected to a structure are infrastructure or infrastructural facilities.

- *A structure's operations represent the total of entities employed to undertake activities or work. A structure's operations represent a structure's operative function.*

When it concerns building architecture, everyone immediately has a certain image of a structure's decorative functions or of the decorative functions of a structure's entities. When it concerns an enterprise, an image is not often perceived automatically, despite an enterprise's certain decorative entities. Take for instance a program showing an enterprise's identity, motivation workshops, the manner in which workplaces are furnished and the image produced on websites of how an enterprise or information system is structured, segmented or operates.

- *A structure's decoration is the total of all entities that do not have current meaningful constructive or operative functions, but displays atmosphere, emotion, feeling and perception with the structure's users and stakeholders.*

Because an enterprise can be compared to a structure, it enables the architect to distinguish a structure's aspects and dimensions as singular and separate entities. By taking them singularly the architect is able to design them as separate entities and in theory create an enterprise that's constructively, operatively and decoratively attractive.

An architect designs integral business and IT solutions by means of architectures, described as structures, often consisting of business processes and information systems divided across different domains. The architect must create at least five models in order to provide the client with a maximum of solution possibilities:

1. A structure concept model
2. A structure element model
3. A structure component model
4. A structure object model
5. A technical structure product model

Additionally the architect creates combinations and integrations of the above mentioned structure models. The architect also includes various visualizations and perspectives, such as a combined and integral view of the environmental entities and the constructive, operative and decorative entities of a model. This combines together to communicate with stakeholders and to test if the solutions will lead to the preferred model. For each model the architect also includes a meta model

If a structure or solution includes definable, reusable composite parts of elements, components, objects, and technical products, it pays to recognize these as building blocks. A building block enhances the reusability of entities and hides the complexity behind a building block. Needless to say, that a building block must be carefully documented.

4.5.3. Elements, components, objects and products

Dragon1 recognizes different types of sections within an enterprise or a yet to be designed structure. An architect refers to sections of an enterprise or a yet to be designed structure as entities.

From a generic point of view Dragon1 recognizes the following entities as being parts of an architecture: systems, structures, phenomena, domains, concepts, elements, components, objects, artifacts and technical products. Not all entities within a solution are however important to a structure's architecture. The architect may provide, by way of visualization, such as a drawing, sketch, diagram, collage or image which parts belong to an architecture and which environmental parts of the structures are added for the purpose of creating an image.

Dragon1 utilizes terminology such as UML (Unified Modeling Language) and BPML (Business Process Modeling Language). When Dragon1 does not follow UML and BPML definitions it will be mentioned as such. Various types of models and diagrams have been adopted, as well as expanded, especially at a conceptual level to allow the architect more freedom of design and control.

Dragon1 recognizes five types of entities that play a major role in the design of an architecture structure. It concerns the entities: concept, element, component, object and technical product.

Example of the cooperation between elements and components

The concepts 'client-server-computing' and 'server-based-computing' consist of elements such as 'server' and 'client. In these concepts many clients provide commands to one single server. The server performs related tasks regarding these commands and sends results or information about the results back to the client. By clustering the server's activities the server's processor is even better utilized.

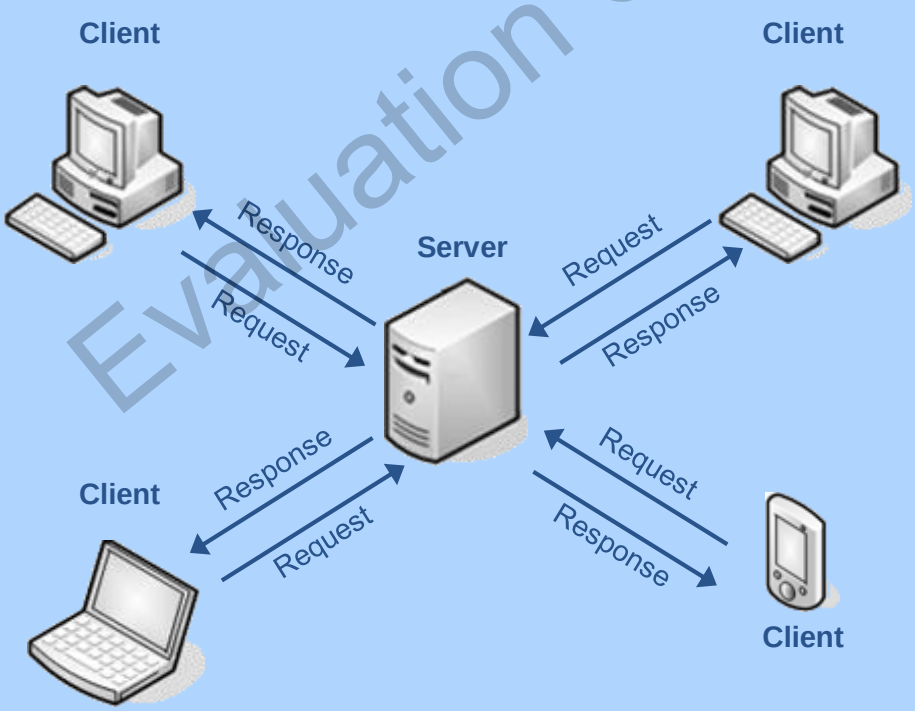

Figure 4.13. Example drawings with different inputs of client-server-computing.

There is a vast difference between client-server-computing and server-based-computing. Server-based-computing implies that almost all activities are performed on a server while the communication between client and server remains minimal regarding: screen based information, keyboard activities, and mouse movements. Client-server-computing involves server processing tasks such as data control. A client is than furnished with the presentation of information and input tasks. The communication at client-server-computing level is far more intense than at server-based-computing level, as masses of data have to be exchanged.

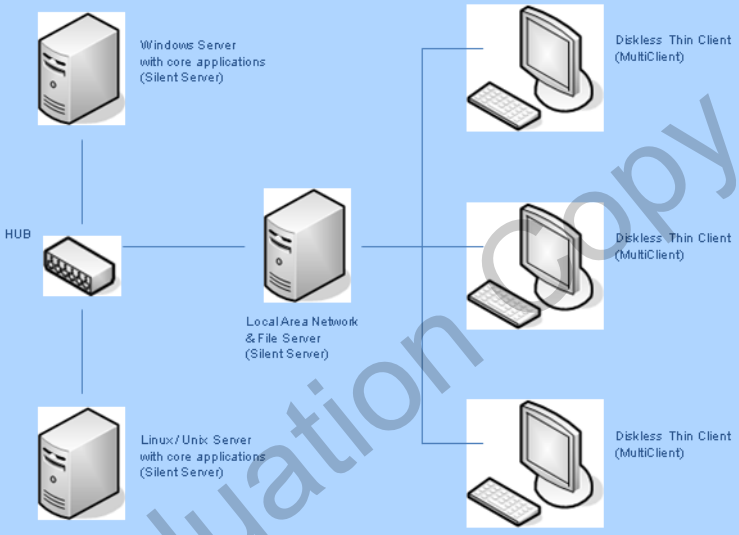

Figure 4.14. Example diagram of server-based-computing: No processing takes place on the thin-client, this happens only on the server.

When the architect however considers a server as the concept, it will contain among others elements, such as: processor, command, data, processing capacity, supply, main panel and memory. The file-and-print-server concept contains the designated elements file and print-jobs. The application-server concept does not include these elements but does however include the application-element.

The above example includes elements of generic concepts, as the elements included are still entities devoid of any technical features: from a technical point of view they are implementation dependent. It has not been decided yet which technology will be used to shape the elements; the architect needs components first to enable him to do so. When the architect takes into account the specific concepts, or the specific elements, implementation dependencies appear and the elements can be denoted as components.

There is a component, referred to as 'diskless thin client', which as part of server-based-

computing, does not include a hard disk but instead constitutes of a screen, keyboard, mouse, and communication software linking network and server. As the architect values the implementation aspect of an entity he does not reference it anymore as an element but as a component. When a server operates on the basis of Windows or Unix, we often refer to it as a Windows' or Unix server. This has also become a component because an implementation aspect has been presented.

An architect can now offer his client an opportunity to choose between two concepts, if he presents the advantages and disadvantages of the two aforementioned concepts side by side, being: large scale cost reduction and manageability (SBC) as opposed to individual freedom toward a client (CSC). The architect uses a concept's operability to clarify how a concept consisting of cooperating elements and components realizes certain qualitative advantages in a given context.

An element is a functional oriented entity and a component is a technical oriented entity. An architect first designs a structure on the basis of elements, in order to translate them later on into components.

Elements possess external features, incorporate a preferred or required image, represent a certain function and demand a particular performance. Components possess external features, incorporate a preferred or required image, and provide a particular function for delivering a certain performance and feature operating possibilities or application possibilities.

For example, in the building architecture a supporting wall and an energy saving water repellent roof are examples of elements. A stone wall and a thatched roof are examples of components. There are constructive, decorative and operative elements as well as components. A constructive element or component takes care of a structure's strength. An operative element or component takes care of operability.

Enterprises and all other systems and domains can be decomposed in terms of elements. This way the coherence and totality of certain necessary functions become transparent and are made clear in terms of components and as such compositions can be created. Also the independence and totality of certain performances become clear and transparent. The system decompositions of elements and components become interrelated in an n:m relationship, meaning that one or more components could belong to one single element and the other way around. An architect will be able to create as such a difference between the design of logical functions and the

technical forms of a structure's part-solutions. This will ultimately provide for a better quality and more attractive structure.

The architect's choice of certain elements and components in a structure is based on the qualitative advantages provided by the concepts in which these elements and components appear. The concept's effect could mean that the implementation of a concept in a certain environment could constitute qualitative advantages. A concept is always designed for a particular purpose in a context or has proved itself as such previously. Some concepts always work in a particular fashion because of external or internal mechanisms. The elements' behavior regarding each other in a concept or their collaboration is inherent to the rules of the concept.

An example of a business administrative concept within a large enterprise is a strict division of tasks and discipline. This concept does not apply to a small business because different tasks are performed within one discipline. In small organizations or sub-organizations within large organizations, business roles often become the operative entity, but this will revert back to strict task and discipline division when other organizations become involved. Business roles enable people to work more flexible and to replace each other. This works well in small informal enterprises where service of a variable quality is more acceptable than in a large enterprise, where a continuous quality of service is required.

Increasing and better qualitative market operation by way of privatization is a principle increasingly employed by public authorities regarding the provision of public services, such as energy, care, transport, and infrastructure. However, the concept of privatization often fails to lead to a properly executed 1-on-1 market service in a new sector. In order to assure proper balance, authorities need to maintain an enforcement mechanism.

Elements such as customers, processes, products, information service, and IT infrastructure are known as concepts in terms of operation, structure and qualitative advantage, that an average architect no longer challenges why they are necessary and if there is no other innovative way in which to solve design questions.

A relatively new enterprise element is a service. From a conceptual perspective this is still an unknown element and is used by architects without the concept having been proved, or the structure, the operation and the advantage having been made clear. Therefore, service is considered more often at a conceptual level.

An enterprise often chooses to work with objects, such as those that handle and process sets of information referring to entities outside the enterprise, such as customers, contract mutations and change of address. A collection of related information that identifies and characterizes a customer, together form the object called, customer. The same applies, when it concerns other related information, such as contract mutation and change of address. To an enterprise, the object - customer - implies a digital or virtual object and is often referred to as an enterprise object, because it is of significance to people in the enterprise.

We consider objects to be a specialization of a component. A digital component or virtual component is in fact an object. A physical customer is a component, customer information is an object. Objects enable the architect to model information about physical components, such as the virtual and digital information world side by side to the physical world. The architect needs components and objects to model the virtual and physical world.

In an enterprise we prefer to avoid having unique essential elements, components, objects or products made with craftsmanship. These entities are called artifacts. Artifacts are non-standard and are expensive regarding maintenance and continuous development mostly due to scarce knowledge and availability of technology regarding the architect.

Examples of enterprise artifacts are: privately constructed information systems, privately invented, undocumented business processes, which constitute a combination of common processes, exotic operating systems or hardware, or a vendor built, bespoke middle-ware software tier (layer). We often see these artifacts in enterprises and real architects will make an effort to expose them with the intention to replace them with style elements and standard components, which offer a much better return on investment and lower operating cost.

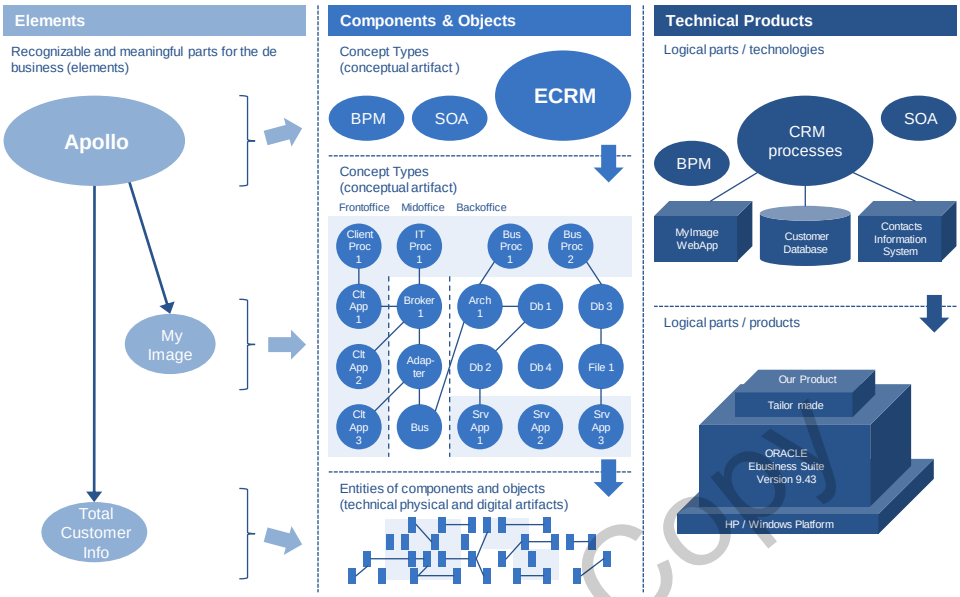

Figure 4.15. Examples of elements, components and technical products.

In an enterprise we notice certain people exercising ownership of elements and components. This may be on the basis of their previous position as shown in an organigram or on the basis of their real position. Element and component groups, by virtue of responsibility and authorization, come under the same control directives are referred to by Dragon1 as domains. We hereby refer to the different types of sales support information systems used by different international enterprise businesses. Together they belong to the information domain, 'sales'.

There are, next to elements (functionally oriented entities) and components (technically oriented entities), also technical products. These are purchased items obtained from a specific vendor. For the realization of the structure, each component must eventually be translated by the architect into vendor purchased or fabricated technical products.

By indicating the difference between concepts, elements, components, objects and technical products, the architect is able to adapt his communication about the yet to be realized structure in accordance with (the requirements of) the various stakeholders, including client and supplier.

4.5.4. Domains

Dragon1 recognizes, as with all other architecture approaches, items such as enterprise domains, because the areas of responsibility and authority are often different from what is shown in an organigram. Domains provide clarity to the in-transparent, parallel and informal part of the enterprise. Altering domain perimeters can invoke resistance, thus by recognizing domains an architect can create a sustainable and realizable design in advance.

Domains allow the creation of chambers within a solution, whereby each chamber can be independently changed internally. Domains enable the architect to design a solution comprising a flexible and resistant construction. An architect could create a domain by defining and furnishing the domain with an owner with responsibilities and authorities.

In reality it often happens that when a business process in a system that is undergoing change, is present in two and more domains. Most of the times this will cause problems to changes, because the architect is confronted with two or more parties having conflicting requirements. The architect hereby shows his ability to solve domain conflicts by for instance designing modular solutions.

4.5.5. Enterprise views and perspectives

Dragon1 differentiates in the way an architect views an enterprise. The architect could view one aspect or many different aspects. If the architect looks at one aspect it is called a side 'view'. If he looks different interrelated aspects at the same time, we call what he sees a 'perspective'. In this paragraph we cover different common side views and perspectives.

Dragon1 model – Enterprise views and perspectives

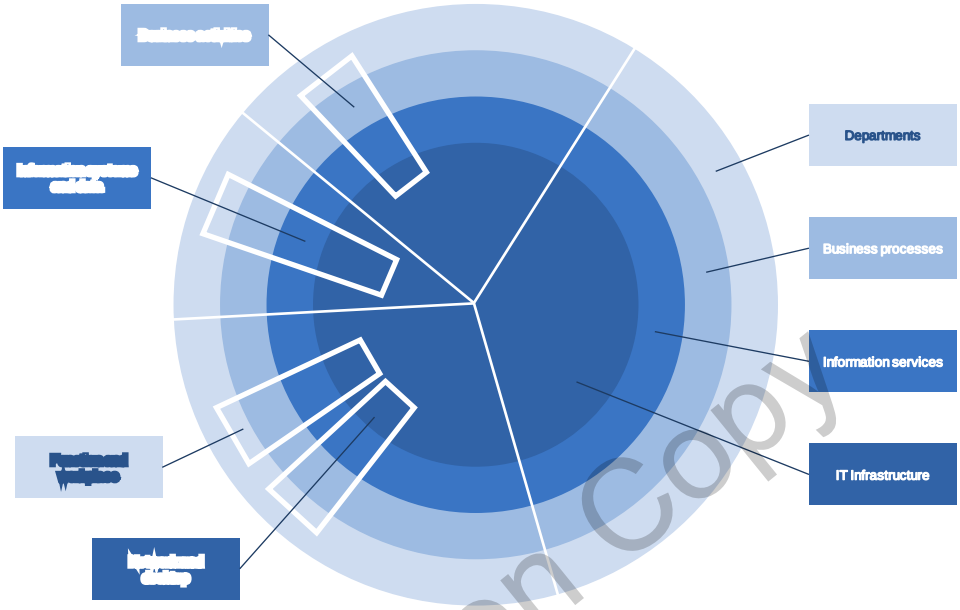

Figure 4.16. Enterprise views and perspectives.

The **Enterprise views and perspectives model** shows an example of which common views of an enterprise Dragon1 recognizes: the business process view, the business function view and the department view. A business process view only shows business processes and their inter-relationships. Business processes, business functions and departments are common business administrative elements, i.e. logical elements which are often applied in an enterprise.

We could also consider logical entities as concepts, in which case we not only look at business processes in the case of business processes, but also at elements and components which constitute the business process concept, such as for instance the business activities. If it concerns viewing different elements and components of a business process we speak of a business process perspective.

Similarly, we speak of an information system view, and information system perspective. An information system consists of the total of applications, people and documents, consequently it consists of different aspects that can be viewed and therefore they also become visible.

Some entities only appear once in an enterprise, such as information provision and IT infrastructure. In these cases we always emphasize a perspective, i.e. the entities that constitute information provision and IT infrastructure. If the enterprise is considered complex we could create an IT infrastructure view, that would mean getting different IT infrastructures into view, side by side, which might result in a complex arrangement. However, by creating this view the complexity becomes more plan-able, controllable, and predictable. The common enterprise perspectives are further described in the following paragraphs.

Business process perspective

An enterprise business process perspective allows the architect to see organization, structures, and decomposition of the enterprise's activities in terms of business processes, work processes, process steps, activities, tasks and actions. The architect becomes aware of how work is organized in the enterprise. This produces a sum total that is not bound to any organizational format or certain competences. This way the enterprise can be designed independently from departments and staff.

Business function perspective

The enterprise business function perspective allows the architect to see organization, structures, and decomposition of business functions and staff. This produces the sum total of requirements in terms of knowledge, ability, skills, and competences to enable work to be executed in the enterprise. In reality the business function perspective and business process perspective are linked together. The architect must however allow the business process perspective to lead, ahead of the business function perspective. The reason is that, regarding the way something is being executed, a business process constitutes a more concrete organization by comparison to a business function.

Department perspective

An enterprise department perspective allows an architect to see organization, structures and decompositions of locations and spaces where work is being executed. This perspective is currently often omitted, although it is nevertheless important to view the shortcomings of the current and future organizational situation. This perspective must however not lead, ahead of the business process perspective. A department is an even more concrete organization by comparison to a business process.

Business IT perspective

In addition the aforementioned perspectives, an enterprise acknowledges a new perspective: the business IT perspective. This perspective shows the dependencies on activities such as conducting business, dealing and working inside the businesses of the enterprise. This perspective provides a decomposition of integral business IT solutions in the enterprise: which business activity is dependent on which IT activity and what does IT contribute to innovation.

This can be an eye opener for many people. If an architect can make it clear to the owners of business processes to which degree their processes depend on efficient operations of information systems, they are often taken aback by the explanation and usually take more notice of the importance of enterprise IT operations.

Operating model

Due to the increased interest in the 'operating model' we hereby note that an operating model comes into existence when the aforementioned enterprise perspectives are all modeled together into one.

4.5.6. The integral Business IT solution as architectural structure

This paragraph focuses in detail on the integral business IT solution. Dragon1 provides examples that prove that much can be gained if an architect regards the integral business IT solution as an architectural structure.

Business IT alignment

An integral solution is the perfectly aligned total of business and IT entities. The perfect alignment is called business IT alignment. The architect needs to be aware of an enterprise's mission, vision and objectives, in order to achieve an optimal alignment between business and IT entities. The architect will then be able to suitably innovate at a business and IT level.

Dragon1 model – Business IT alignment

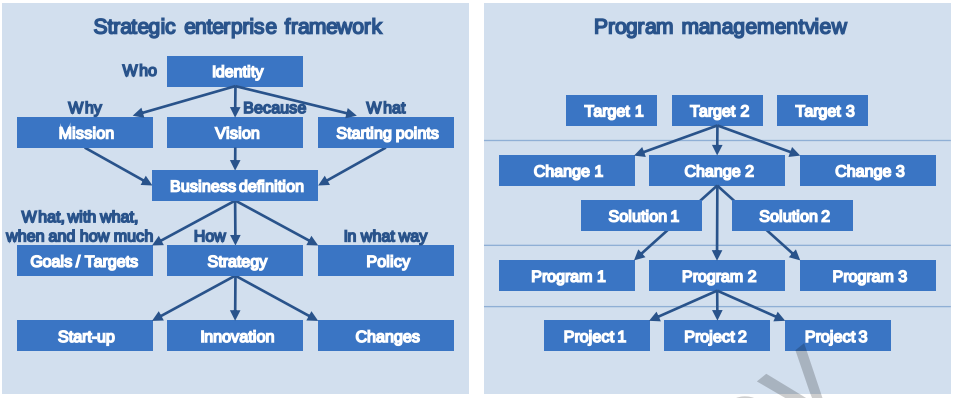

Figure 4.17. Business IT alignment.

The **Business IT alignment model** provides an example of how strategy ensues from mission, vision, and objectives and how strategy directs innovation. This entails that new IT options should first lead to new perspectives at board and directors' level, ahead of adapting vision and objectives in which new IT options are used. This alignment model enables business management to remain in control of IT and IT remains in support of management.

To achieve the realization of objectives, changes are sometimes needed. For this purpose organizations will use certain instruments, programs and projects. A proper relation and breakdown of objectives are pre-conditional to assure the integrity of required changes, solutions, programs and projects. This will ensure that only approved IT options are used within missions, visions and strategy. Figure 4.17 shows which relations are required by a business to control and manage IT and not the other way around.

Additionally, the architect needs to package his design through a program management perspective, to indicate which parts of the design, in which program and project, must be developed separately or integrally. Herewith, the architect assures that the risk of design failure is reduced when part of a solution is not realized in time.

CTI concept

Dragon1 takes an entirely new approach to IT, because Dragon1 regards the integral business IT solution to be an architectural structure. By looking at IT in an abstract and analytical way, it can be ascertained that IT enables an enterprise to inform,

communicate, collaborate and undertake transactions. Infrastructures are used to provide the necessary resources, and as such IT becomes CTI.

- *CTI stands for Communication & Collaboration, Transaction and Information. The resources that are used are called CTI resources and the applicable infrastructures are CTI infrastructures.*

It is currently acceptable to speak of information facility and state that these are supported or made possible by means of IT infrastructure. In turn CTI enables the architect to design and integrate communication, collaboration and information resources separately.

In addition Dragon1 recognizes an enterprise to operate common facilities at all levels and that there are generic, specific, as well as packaged and non-packaged facilities. By offering an architect these degrees of freedom in facilities and infrastructure, he becomes less restricted in his freedom to create an architecture design.

When a user has no or only partial control over facilities, we speak of infrastructural facilities. To all intents and purposes infrastructure covers the sum total of immobile facilities present. For instance an ATM is an infrastructural facility however a mobile hotdog stand is not, although it is a facility of sorts.

- *An (information) facility is the total of an information system that offers certain functions. The information provision consists of a total of resources of which the user has no, or only partial control.*

Multi-channel FMB-office concept

Many enterprises operate a tripartite system regarding, communication, transactions and information because of the nature and dynamics involved.

An enterprise consists of an internal dimension for the purpose of processing and an external dimension for the purpose of interaction. Where there is information processing, the enterprise requires the highest of stability through heavy duty, robust systems. The place where this happens in an enterprise is known as the back office.

- *A back office is an enterprise domain which focuses on large scale data processing. The enterprise wants the concept of operational excellence to apply in their back office. Customer interaction is not envisaged or needed.*

Where there is extensive interaction with the outside world and customers, the enterprise wants to display its ability to be flexible and customer focused. The area where this takes place in an enterprise is called the front office. Lately front offices are often the subject of communication channel renovation, in order to offer channel transparent products and services and communicate with their customers.

- *A front office is a dynamic enterprise domain which maintains contact with customers and as such focuses on the concept of customer confidence.*

Looking at front office activities, communication channel problems do not interfere with the heavy data processing activities of a back office, as their various multimedia communication channels operate independently.

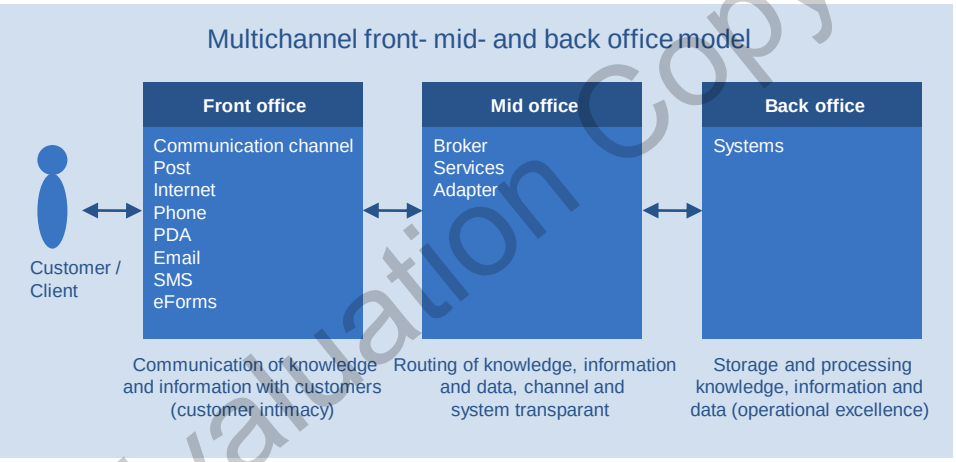

Figure 4.18. Example of a multi-channel front- mid- en back office.

Figure 4.18 shows an example of an office domain classification, which is in conformance with the FMB-Office concept. It introduces an enterprise construction, which enables innovation to take place within the domains of an enterprise in a much simpler way. Changes introduced to the back office would not affect front office activities, or the other way around.

The mid office ensures that the back office communication channel can continue to operate in a transparent way. Communication, transactions and information are offered in a channel transparent (independent) way to back office systems. The mid office ensures that in the front office, independent of the back office technology, they can independently communicate with the customers, transactions can take place and

information can be supplied. Information, transactions and communication are technology independently offered to the front office systems.

- *A mid office is an enterprise domain that controls the interaction, routing and (data) communication between front office and back office and maintains contact with other mid offices.*

The advantage of operating a mid office is that the number of control links between front office entities and back office entities get dramatically reduced and with that maintenance costs. As a consequence changes in the enterprise can now be applied more quickly. A disadvantage of a mid office is that the knowledge base and IT are intensive and consequently require more maturity, a higher realization budget and directing of the enterprise's IT organization.

4.6 Architecture design

An architect's most important product is the architecture design of a structure. In the creation of architecture designs lies the essence of an architect's work. This is what it is all about. In the architecture design, the architect makes the conceptual translation from strategic starting points, objects, requirements, and the client's preconditions, via concepts and principles to realizable and affordable solutions.

A program of requirements always provides the basis for the choice of the decomposition of the entities and the relations and the entities of a structure. An architecture design interconnects constructive entities, operative entities and decorative entities in a harmonious way, to accomplish a desired atmosphere, emotion, feeling and experience in a structure.

The entities described and visualized in an architecture design usually comprise: (sub) domains, (total) concepts, part concepts, phenomena, (part) systems, (style) elements, components, objects and technical products. Following this, entities become specific by translating them in accordance with the principles, rules, regulations, directives, laws, norms, standards and values applicable to entities.

By configuring entities the architect will attempt to comply with, and give substance to strategic starting points, objectives, requirements, and preconditions of the client and stakeholders, constructively, operatively as well as decoratively. The situations, spaces and locations must ensure the appropriate use of system configurations in conformance with quality aspects such as sustainability, future proof ability, comfort and ease, robustness, adaptability, and target oriented.

An architecture design consists of four consecutive levels of detail:

- Conceptual architecture design (CAD) – the architect specifically recognizes concepts, principles, rules, and style elements. More choices are possible here.
- Preliminary architecture design (PAD) – the architect recognizes mainly elements and rules from concepts and principles. There are less choices and variations on a scenario available here compared to CAD.
- Definitive architecture design (DAD) – the architect recognizes principally components and objects.

- Technical architecture design (TAD) – the architect recognizes principally technical products.

These four parts of an architecture design become consecutively iterative with the help of functional specification. In every part there is an analysis of a current situation and an architecture design of a desired and required situation, sometimes divided over different architecture plateaus and detailed into various scenarios.

CAD represents the conceptual abstraction level of an architecture design or is a design description of the total concept of a structure. To all intents and purposes the total concept of a structure is the architecture of a structure. PAD and DAD constitutes the logical abstraction levels or the logical architecture levels and the TAD the physical abstraction level or the physical architecture design.

CAD contains mostly informal sketches of concepts and principles, storyboards of scenarios and personas in an enterprise and the stakeholders, who globally indicate their preferences of where it all should end. The elements and components are already globally described and sketched. The total solution sketch is a perspective that reappears as standard visualization in CAD

PAD contains mostly drawings with more formality and details than sketches. PAD comprises elements and components and there are more choices than in CAD. Cartoon drawing, design sketch and design drawing are standard perspectives of PAD.

DAD contains mostly large global overview diagrams, and very formal visualizations, which serve as contract documents for other parties due to their accuracy. Blue prints, artist impressions, principal details and visions of structures are examples of perspectives, which reappear as standard visualizations in DAD.

TAD contains part diagrams with many small details, which are used as construction drawings by engineers to solve design questions. Whatever can be specified and visualized at a technical level by assembly takes place here.

An architecture design contains visualizations of perspectives of models, whereby perspectives interpret parts of models perspectives. Take for instance, the design sketch of the profit making perspective of a future financial model of a marketing and sales business, which shows why and where profit is made, whilst the financial model itself does not show it.

Example of a translation in an architecture design

The client and stakeholders of an architecture design for a new information system want to market information products, such as research reports with research details. The customer will be enabled to compile his own report on the basis of different parameters whereby business managers decide on a daily basis which combinations and parameters of figures (rates) are feasible. In addition the background, the back office systems, must be opened up to the front office, to allow the customer to browse the internet for information, by accessing information held at the back office to assess which report and figures (rates) he requires.

The architect must decide which concepts of the information system he is going to use to enable customized information products to be offered safely and speedily and payment to be settled quickly. Also how he is going to make windows to accommodate safe real time browsing of back office systems.

The architect thinks in terms of a copy-production-database concept to construct safe window functions, a business-rules-engine concept and the online-production-configuration concept to enable managers and customers to compile their own products.

The next step is that the architect describes and visualizes the principles and functional elements of these elements. The following step is to recognize and name the technical components from which elements are realized. He then proceeds to search for suppliers who supply product solutions for the recognized components.

Eventually the architect will ideally arrive from concepts to technical products that can be bought off the shelf from a store.

- *An architecture design is a design of a total concept for a structure which consists of an analysis of a current configuration, space, environment, location, and situation of an existing system and a design of a future configuration, space, location and situation. The configuration, space, location and situation are made visible and controllable with descriptions and visualizations of entities at a conceptual, logical, and physical level. In an architecture design the translation is made from concepts to elements, from elements to components and objects, and from components and objects to technical products.*

Dragon1 model – Architecture design meta model

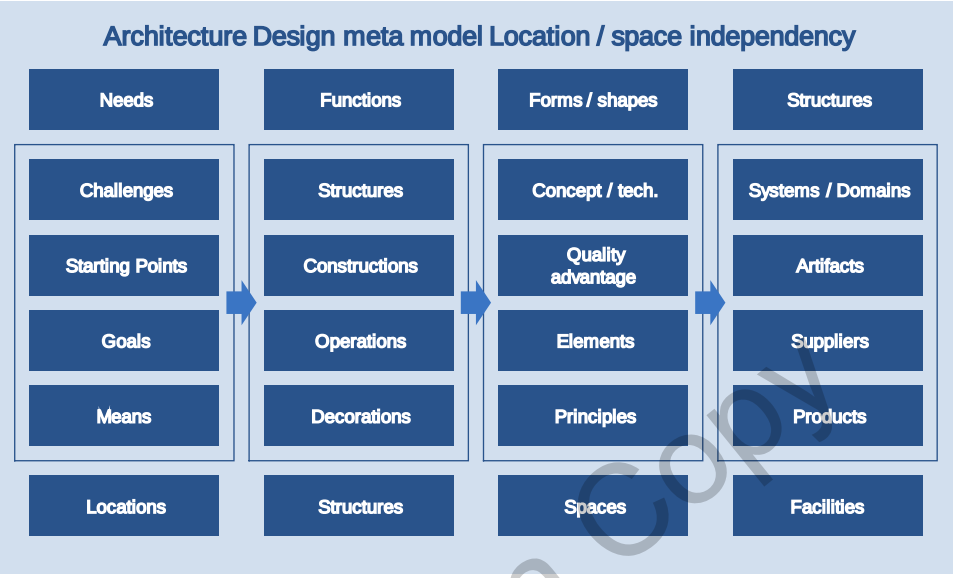

Figure 4.19. Architecture design.

The **Architecture design meta model** shows an example of the most important types of entities that constitute part of the architecture design. The architect translates needs, challenges, strategic starting points, objectives and requirements into structures, constructions, operations, and decorations as an integrated part of a structure. The architect will then detail the construction with concepts, elements and principles. The structure is foreseen at different locations with spaces in which people can utilize certain facilities. The elements are translated by the architect into components and technical products. All these aspects reappear in different parts of the architecture design.

Parallel to an architecture design a master plan will be prepared by the program management and the architect. This master plan shows the different phases or levels along which the architecture design will be detailed and realized. To realize an architecture design, programs and projects are usually started up and when more details are added to the master plan it will be further adapted and detailed.

The architecture is to all intents and purposes an attachment of a master plan. In reverse order it can be stated that in each plan there will be always be an architecture design implicitly present. By making this explicitly noticeable and complete, more control and management becomes possible regarding risks and program quality.

- *A master plan is an overall planning of all activities that are eventually realized by an architecture design.*

4.5.1. Chain architecture design

When different enterprises and their businesses collaborate to supply the customer with a product or service in a quicker and more proficient way, they could agree to establish a service supply chain.

The chain architecture design consists of models such as the chain configuration model, a chain domain model, a chain function model, a total concept and principle model and a chain meta model. Usually these models are constructed in global and then detailed form. A chain usually consists of various enterprises, businesses and processes, from which generic and detail models are frequently made. The chain's directors and managers can utilize such models to hedge the entire chain with specific links. Different links, which are organizational units in their own right, can be used for those models which are a link in the chain.

The chain architecture design makes it clear how the architect uses certain concepts, at a conceptual level, to complete strategic starting points, preconditions, requirements, and objectives put forward by the chain's stakeholders. This could be for instance a chain, which can provide "a business-oriented" service, but also a chain which can be centrally and group managed by representatives of the most important links in the chain.

In the chain architecture design, elements reappear such as link, business, critical path, case and service. The design incorporates the previously used principle: 'the weakest link in the chain determines its strength'. And, a rule such as: 'do not compromise the critical path by other work activities on a case'.

- *A chain architecture design is a design of a chain of businesses, enterprises or institutions, with a focus on the qualitative realization of a total concept of a chain. In a chain, processes, information service and IT structure are recognized as an integral part of the individual business links.*

4.5.2. Enterprise architecture design

When different smaller enterprises decide to collaborate as a single large enterprise, enabling a more powerful presence on the market, the implementation of the merger has to be supported by an architecture design, showing the collaborative potency of the merging enterprises. An enterprise architect provides visibility of the type of enterprise to be designed and which architecture styles will be used in the design. An enterprise architecture design is usually adorned by a catchy title and a time epoch, for instance: NewVenture 2012

An enterprise architecture design consists of models, such as an enterprise architecture configuration model, an enterprise function model, an enterprise (system) domain model, a total enterprise concept and principle model and an enterprise architecture meta model. The models are derived from overarching meta models. A function model usually sports an available decomposition.

Dragon1 model – Enterprise (system) domain meta model

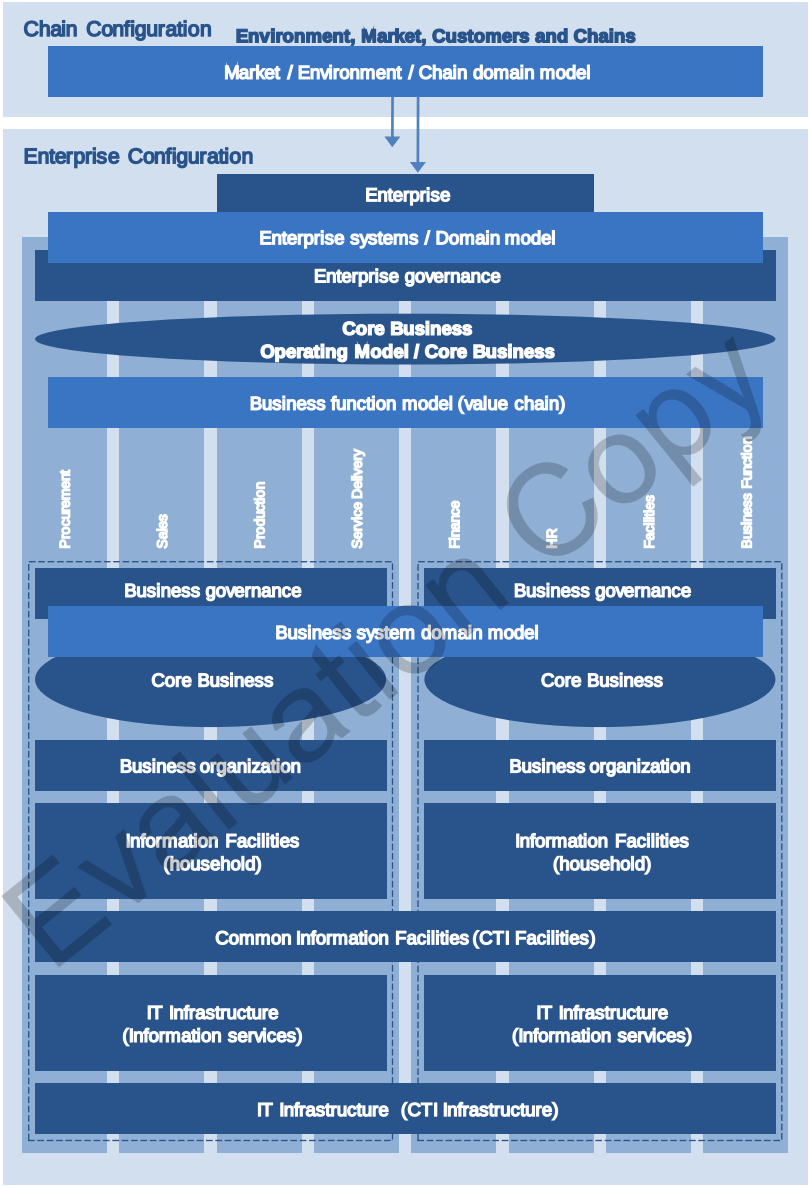

Figure 4.20. Enterprise (system) domain.

The meta model **Enterprise (system) domain** includes an example of how systems and domains are increasingly recognized in enterprises in a particular way. These systems and domains display generic, frequently occurring entities.

When an enterprise is considered as a structure, systems and domains are understood to be enterprise-wide building parts, whereby reused, compounded entities can be recognized as building blocks.

A business function is therefore rather a building part of an enterprise structure, the business process and the business activity are then building blocks and a handling becomes for instance an entity.

Dragon1 model – Enterprise configuration meta model

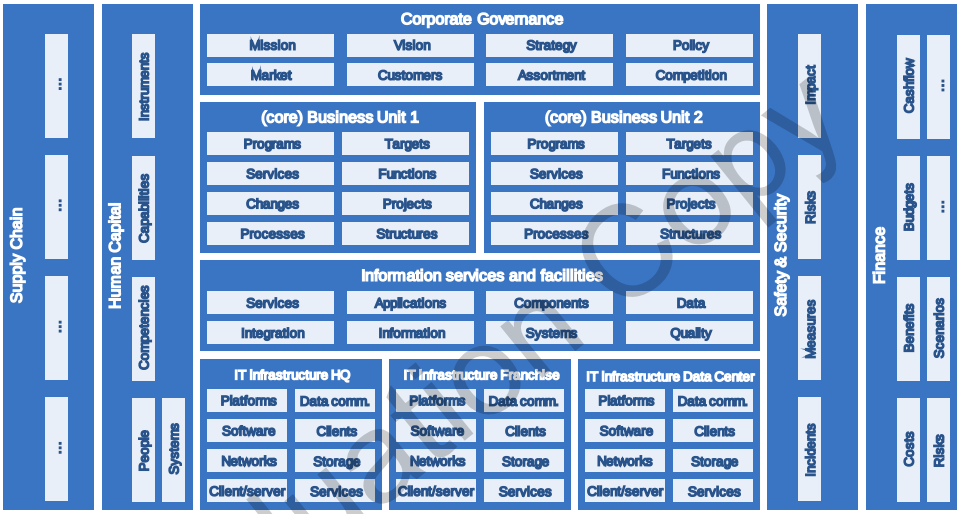

Figure 4.21. Enterprise configuration.

The **Enterprise configuration meta model** includes a generic example of enterprise-wide entities, which often appear in enterprise systems and domains. These entities require an architecture design statement regarding concepts, principles, style elements, components, objects, rules and technical products.

An architecture design shows how an architect, supported by certain concepts completes, at a conceptual level, strategic starting points, preconditions, requirements, and objectives of the enterprise in accordance with the input of stakeholders. For instance: design an enterprise which is customer focused and which supplies without exception standard products and standard services to a wide customer base. The architecture design boasts reusable principles, for instance: CRM and an integral customer base require an increase in customer focus, and cross-selling and up-selling. And rules, such as: modular products and services make many custom made standard solutions possible.

- *An enterprise architecture design focuses on appropriate qualitative realization of a total concept of an enterprise, whereby control, the businesses, the information service and the IT infrastructure are important, separately identified entities, in terms of being concepts, domains, elements, components, objects and technical products. In addition the relationship between entities, such as principles and rules are identified.*

4.5.3. Business architecture design

Imagine an business architect who is commissioned to accommodate all IT processes of an enterprise in a new Shared Service Center, which has to become a new business within the enterprise. All other businesses are 'compelled' to become customers of the new SSC, which represents the new IT business. In his business architecture design the architect shows which concepts, elements, components and products he has selected to accomplish a smooth running, future proof and sustainable IT operation within the enterprise for a period of five years.

A business architecture design consists among others of the following models: a business configuration model, a business function model, a business domain model, an operational business model, a total business concept and principle model, a business meta model and an architecture and design model. Also present in the business architecture design are various models pertaining to situations, space, location, and environment. The models are derived from overarching meta models.

The business configuration model includes an example of important entities which constitute parts of a business of which a business configuration can be designed.

In the example of a SSC, the architect will possibly recognize a principle such as: 'standard service delivery ensures less costly and less complex service delivery'. And, a rule, for instance: 'all enterprise businesses are obliged to obtain their IT services from the SSC'.

Dragon1 model – Business configuration meta model

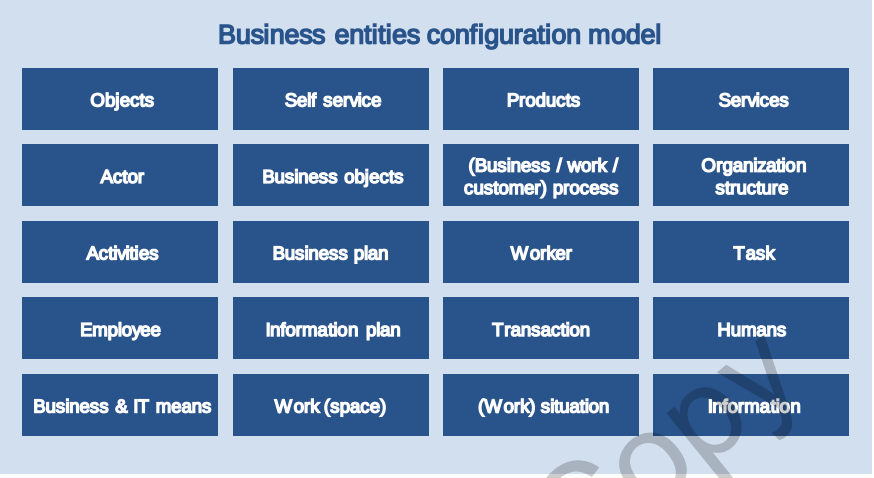

Figure 4.22. Business configuration.

Programs and projects of the various collaborating businesses will benefit from the business architecture design in terms of the realization of the new shared services centre. The principles, style elements, components and rules displayed in the business architecture design compile to constitute the binding frame work for all. A few parts of the design have already been sufficiently elaborated on to be implemented. Other parts must still be designed in more detail before they can be realized. These functional and technical detail designs constitute attachments to the architecture design.

- *A business architecture design focuses on sufficient qualitative realization of a total concept of a business whereby control, parts of the business, business functions, business processes, information service and IT infrastructure are recognized as separate parts in terms of concepts, principles, domains, elements, components, objects, technical products and rules.*

4.5.4. Information architecture design

When the information facility of an enterprise must be renewed, for instance to enable increased digital service and custom made information products to be offered, it is advisable to support these intentions by way of an information architecture design.

An information architect is given a commission to provide a service oriented architecture design for the 'Service Information House (SIH 2.0) by 2014'. The information service is given a type and personal name to make the enterprise uniquely identifiable.

An information architecture design consists among others of the following models: an information configuration model, an information function model, an information domain model, a total information concept and principle model, an information facility meta model and an architecture and design model. When there are a number of models it is considered advisable to support them with a hierarchy model, which shows the relationship between the different models. The models are derived from the overarching meta models.

Dragon1 model – Information facility configuration meta model

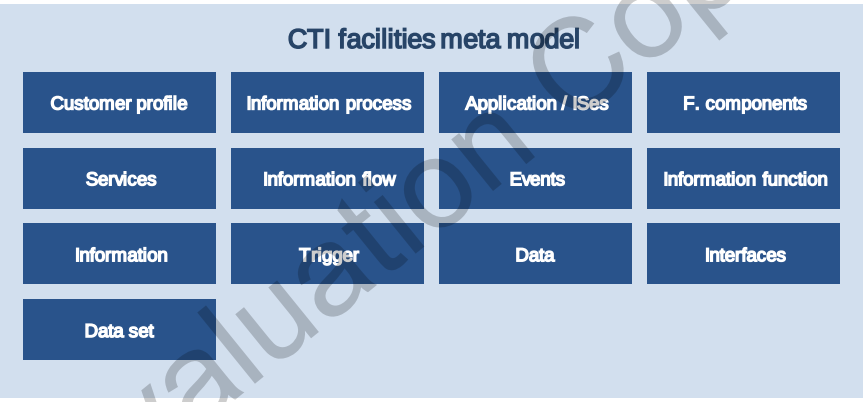

Figure 4.23. Information facility configuration.

The **Information facility configuration meta model** shows an example of entities, which in generic terms always appear in information supply issues and must be completed by the architect.

The information domain model of an architecture design often includes functionally oriented groups of information systems, in terms of business processes, products, services and business functions. For instance, the information domain 'licenses' at municipalities is a collection of applications, and a collection of information dedicated to processing licenses.

The architecture design for SIH 2.0 probably includes the principle: 'the assortment wide supply of digital information products result in satisfied customers'. And the

rule, for instance: 'all data will be filed as close and originally as possible to the source, to achieve the highest quality data storage'.

An architect in his design decision chooses to include a vendor product standard for data storage. He could also have opted for a technology standard for data storage, whereby the engineer of the design is free to make a choice of which supplier will provide the data storage solution. The architect will produce options of strategic importance always for further deliberation and decision by the client.

These information architecture design principles and rules ensure existence of useable frameworks for engineers who participate in the detailing and design of all parts and must ascertain the realization of these parts on a separate basis.

The information architecture design allows the architect to continue beyond the inclusion of principles, concepts and rules. He can go as far as stating, in a substantiated and justified manner, which design patterns must be used, or which standard solutions should serve as a basis for solutions. To what extent this will be detailed used in the architecture design depends on the quality assurance the client requires concerning the solution.

- *An information architecture design focuses on sufficient qualitative realization of the total concept of an information facility by which control, information processes and systems, data gathering and applications and the interfacing between applications is important let alone the identified parts, in terms of concepts, principles, domains, elements, components objects, technical products and rules.*

4.5.5. Technical architecture design

When the IT infrastructure of an enterprise requires renewing, for instance to achieve higher quality customer focused performance and availability of information, it is advisable to create a technical architecture design. The technical architect could for instance be given a commission to create a virtual high performance architecture design for VirtualInfra Now 2014. The IT infrastructure will then be given a generic and personal name to establish extra recognition in the enterprise.

A technical architecture design consists among others of the following models: an IT infrastructure configuration model, an IT infrastructure function model, a technical

domain model, a total technical concept and principle model, an IT infrastructure model and an architecture and design model.

When there are many models present, it is advisable to create a hierarchy model showing the relationship between the various models. The models are derived from overarched meta models.

Dragon1 model – IT infrastructure configuration meta model

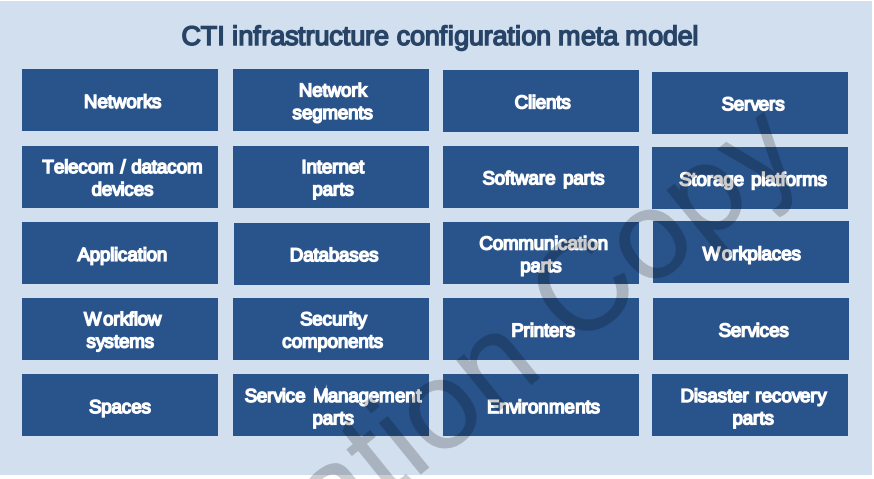

Figure 4.24. IT infrastructure configuration.

The **IT infrastructure configuration meta model** includes a number of entities which in the generic sense, always appear in an IT infrastructure and which must be completed by the architect in his IT infrastructure architecture design.

The technical domain model in the architecture design is often a technically focused group of IT infrastructure components in terms of business processes, products, services and business functions and information facility. For instance, workflow is a technical domain, consisting of a collection of Municipality based technical infrastructural systems, standards and protocols, such as email and SMTP, which promote collaboration between parties.

The technical architecture design of VirtualInfra Now, probably includes the principle: 'virtualization of network, servers and applications contributes to a reduction in complexity and cost'. And, for instance the rule: 'network equipment is always supplied by the same vendor and operates on the basis of international standards and protocols, admitting remote configuration'.

An architect decides to include in his design the choice of using a vendor's product standard for network security, or he could choose a technology standard for network security, whereby the choice of vendor is left to the decision of the engineer of the design. The architect will leave the ultimate strategic decisions to the client, indirectly covering the choice of flexible types of applications, such as service oriented application levels or monolithic applications in his hands.

Such architecture design principles and rules ensure useable frameworks to be incorporated by engineers, who will design further details in a way that all parts are realized separately.

The architect is now able to design beyond just principles, concepts and rules. He is in a position to argue and justifiably state which design patterns need to be used, or which standard product solution must constitute a basis for solutions which are yet to be realized. How far an architecture design will be detailed depends on the degree of assurance the client expects regarding the quality of the solution to be realized.

- *A technical architecture design focuses on sufficient qualitative realization of the total concept of an IT infrastructure facility by which control, technical processes and systems, network and clients, servers and security and management of the IT infrastructure are separately identified parts, in terms of concepts, principles, domains, elements, components objects, technical products and rules.*

4.5.6. Solution architecture design

If an enterprise requires a sustainable and future proof total solution, for instance a coherent combination of business processes and business systems, it is advisable to create a solution architecture design. A solution architect is for instance commissioned to design an architecture entailing an e-CRM solution for an enterprise by 2014. The structure is given a type and identity name in order to be extra identifiable in the enterprise.

A solution architecture consists among others of the following models: a business process configuration model, an information system configuration model, a solution domain model, a total solution concept and principle model, a solution infrastructure meta model and an architecture and design model. When there are many models present, it is advisable to create a hierarchy model showing the relationship between the various models. The models are derived from overarched meta models.

Dragon1 model – Solution architecture design meta model

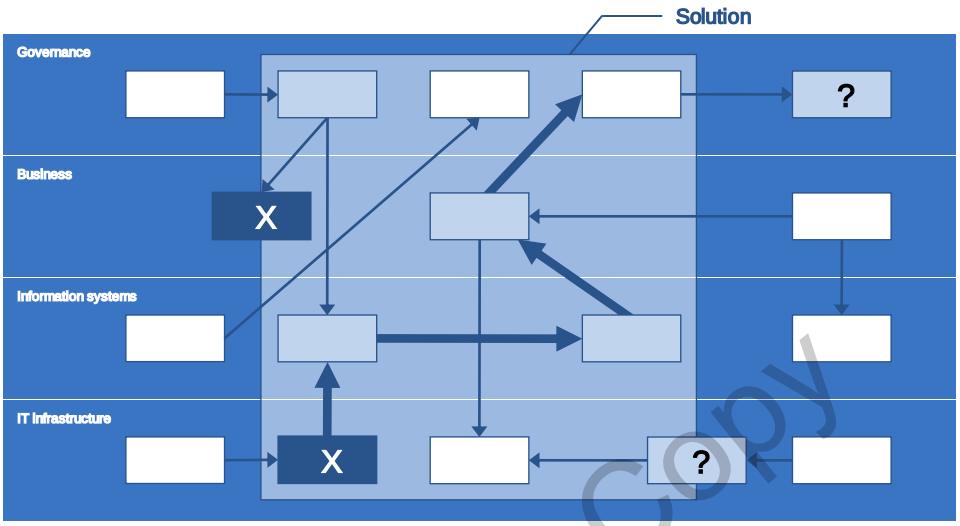

Figure 4.25. Solution architecture design meta model.

The **Solution architecture design meta model** includes an example of why a solution is a structure, and is included in an enterprise which in turn constitutes the context. A solution is not an independent aspect in an enterprise, but it is linked to other entities in different systems and domains of the enterprise. The structure itself is an integral business IT structure, but it must be made an integral part of the enterprise. As far as the enterprise is concerned a solution is a building block.

The solution architecture design meta model includes entities which in a generically sense, always appear in a solution and which must be completed by the architect in his solution architecture design. The solution domain model in an architecture design is often a functional or logically oriented group of business processes and information systems. Imagine a solution as an integral e-CRM solution, the solution would then consist of sales processes, sales systems and the relationship between these two entities.

The solution architecture design of e-CRM probably further contains the principle: 'an integral customer profile simplifies cross and up selling and the enterprise more customer-focused'. And, probably includes the rule: 'every customer has a history from the moment there is customer contact'.

The architect will make a design decision to select an e-CRM branch or product standard, whereby the choice of vender is left open to the builder of the design.

The architect will leave the ultimate strategic decisions to the client, indirectly covering the choice of flexible types of applications, such as service oriented application levels or monolithic applications in his hands.

These architecture design principles and rules ensure existence of useable frameworks for engineers who participate in the detailing and design of all parts and must ascertain the realization of these parts on a separate basis.

The solution architecture design allows the architect to continue beyond the inclusion of principles, concepts and rules. He can identify entities, but also fragments and building blocks of entities, as well as paper models containing entities. He can in a substantiated and justified manner show which design patterns must be used as a building block or fragment, or which standard solutions should serve as a basis for solutions. To what extent this will be detailed used in the architecture design depends on the quality assurance the client requires concerning the solution.

- *A solution architecture design focuses on sufficient qualitative realization of the total concept, usually of business processes and information systems, whereby processes, activities, products, services, structure, applications, databases, links and networks become separately identified parts, in terms of concepts, principles, domains, elements, components objects, technical products and rules.*

4.7 Questions

After studying this chapter the reader should be able to answer the following questions.

4.1. Which are Dragon1's eight concepts of the 'way of thinking'?

4.2. Which types of concepts and principles exist in Dragon1?

4.3. Give an example of a concept and an example of a reality principle.

4.4. Describe what types of entities play a key role in the architecture design of a structure.

Chapter 5

Dragon1 Way of Representing

The communicative visualization of enterprise architecture
in support of making decisions
based on insight and overview.

5.1 Study Objectives

After studying this chapter the reader the reader is able:

- To describe which concepts constitute the Dragon1 Way of Representing.
- To describe which types of architecture visualizations there are.
- To describe which types of structure models there are.
- To describe the viewpoints, views and perspectives of these models.
- To describe how an architect designs an architecture poster.

5.2 Introduction

The chapter 'Way of Representing' elaborates on the third part of Dragon1. The 'way of representing' focuses on architecture visualization. By using models and looking at different views, the architect will work out various perspectives of a structure to show the possible effects and consequences of the requirements issued by the client, stakeholders and users. For this purpose the architect has at his disposal various types of architecture visualizations, among which an architecture visualization-poster, incorporating a standard view-layout and visualization design principles.

In the 'way of representing' Dragon1 brings to focus all aspects of defining, visualizing, analyzing and designing structure perspectives, as the architect's principal discipline. It involves getting the architect to bring into view important aspects for which choices need to be made; aspects which require insight and an overview of interdependencies and connectivity, or aspects which require an impact analysis of change.

To achieve sharpness of image to discover appropriate viewpoints, views and perspectives it is important to reach a proper understanding of different types of architecture visualizations, as well as finding the right kind of models acceptable for enterprises and viewers, the group of stakeholders for whom visualizations are meant. When the architect becomes aware of for what, for whom and why for an architect visualization is meant, the information can be molded in an optimal way and the message enhanced to tune in with viewers.

When constructing architecture visualizations, the architect consequently focuses on the viewers for whom the visualizations are intended.

Dragon1 distinguishes three important techniques: the creation of models, the creation of perspectives and the creation of architecture visualizations.

5.2.1 Way of Representing Framework

On the whole, the 'way of representing' can be described by one architecture framework. The framework consists of eight 'way of representing' concepts. Principally the way of representing questions: how the architect is to take into account the different ways of modeling and visualizing, as well as how he can make use of the different kinds of architecture visualizations.

Dragon1 model – Way of Representing Framework

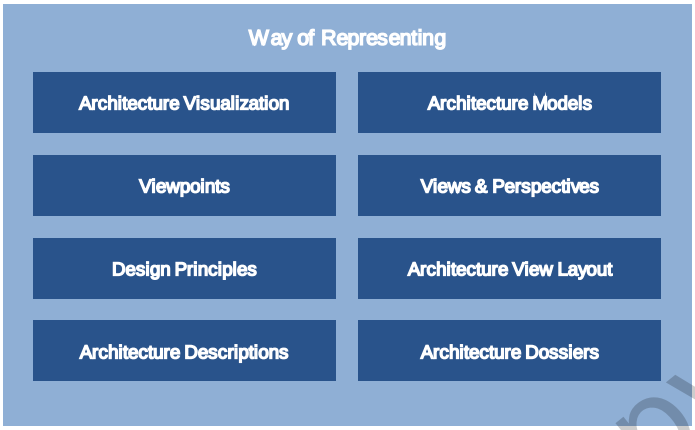

Figure 5.1. Way of Representing Framework.

The concept 'Architecture Visualizations' is the way how by using different kinds of technical graphical models, overviews and perspectives can be represented graphically.

The concept 'Architecture Models' stands how to translate a structure into a structured collection of entities and information.

The concept 'Viewpoints' shows how people tend to view structures and how the architect includes this concept in the visualization of views as seen by viewers.

The concept 'Views & Perspectives' is the way how different types of perspectives of a structure in an architecture design are represented. In turn how different types of perspectives of a structure are communicated to stakeholders, who are as such supported in making informed decisions.

The concept 'Design Principles' is the way how the architect can create pleasant and compelling architecture visualizations, as he is able to take into account cultural filters and the manifestation of what viewer's eyes convey.

The concept 'Architecture View Layout' shows how information can be optimally inserted into an architecture visualization poster, in order for stakeholders to understand the architect's conceptual message more clearly.

The concept 'Architecture Descriptions' covers which standard ways are applicable to architectures, in order that they will be of a high quality.

The concept 'Architecture Dossiers' explains which standard dossiers can be identified for architecture work, in order that information does not become scattered, but rather becomes easily available and therefore functional.

The following paragraphs consecutively cover a number of 'way of representing' concepts from the Figure 5.1 framework, in a vertical as well as horizontal manner. Each 'way of representing' concept is elaborated on by way of one or more Dragon1 models.

- *A representation is the graphical or textual reproduction (visualization) or description of an item such as a system, phenomenon or concept.*

5.3 Architecture visualizations

In previous chapters we stated that architecture design is all about making the appropriate conceptual translation from strategic assumptions, targets, requirements and preconditions via concepts, principles, elements and components to physical and digital technical products. Regarding this the architect will create different decompositions of different aspects of a structure.

To achieve 'sharpness of image' regarding conceptual translations, thereby handing the client significant control over the end quality can be considered a major challenge for any architect. As surely, any decision made by the architect is of significant consequence to a satisfactory and conforming-by-rules adaptation of stakeholder requirements.

The architect makes design decisions, however he must in the event of far reaching consequences present strategic options to the client for approval. In order to evaluate situations and impacts more quickly the architect will prefer to use visualizations over written text to inform his client and stakeholders.

Visualizations, graphical images, are able to communicate certain complex matters and dependencies much faster than a written document. One A0 size visualization poster can easily replace a 100-page document. Architecture visualizations show a certain perspective of a model, taking into account stakeholder viewpoints.

Architecture visualizations carry a title, technical tag and nomenclature in which structure, model, perspective and type of visualization all recur. For instance: the design vision of the new Apollo information system is a principle drawing. Or the blueprint of an existing NewX IT infrastructure is a component diagram. Apollo and NewX are structures of which an information concept model and an IT component configuration model are made. Design vision and blueprint are the names of types of perspectives used for visualizing models. Principle drawing and component diagram are the names given to types of architecture visualizations, indicating the visualization technique that has been used for creating a visualization.

Strictly speaking an architecture visualization is the visualization of architecture, i.e. concepts and principles. In reality the different visualizations of an architecture design are all called architecture visualizations. Dragon1 takes it that verbally this does not cause problems, however when using text it is important to clearly indicate

the difference between visualizations depicting architecture and visualizations which do not.

5.3.1 The role of architecture visualizations

Dragon1 distinguishes different types of architecture visualizations, all of which have their own role. Each role stands for a different decision which can be taken within enterprises.

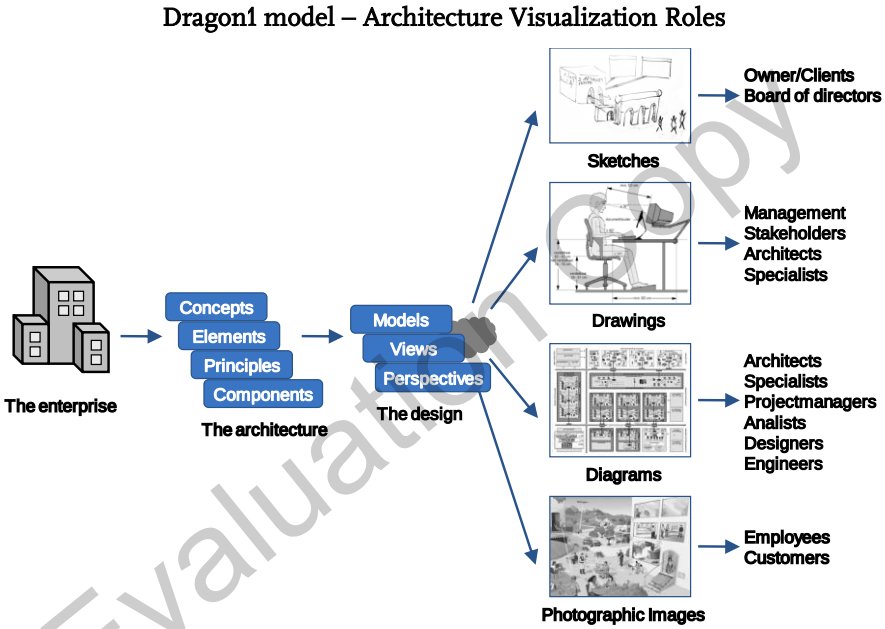

Figure 5.2. Architecture visualization roles.

The **Architecture Visualization Roles model** provides examples of how an enterprise can be visualized as a structure consisting of the sum total of entities, such as concepts, principles and style elements.

Together these three entities constitute the basis of the architecture of each structure. Different perspectives of models and detail models can be made of these entities.

We clearly notice how each visualization implicitly shows the perspective of a model including certain concepts and elements. When a visualization is looked at by us it is imperative that these aspects should be emphasized when they appear in our area of

interest. We notice how the architect can present different aspects to different observers with the same views of a model.

Clients such as board and management, owners and steering committees prefer to use sketches, because they are informal architecture visualizations which can be rapidly produced to concretize certain ideas and thoughts which still contain many uncertainties and unknowns surrounding various issues. If an architecture visualization is required by the Board and directors, sketches often prove to work better than diagrams.

Managers and policy makers who are responsible for issuing contracts to, or making legal commitments with, vendors and implementers require accurate architecture visualization. In these cases it is preferable to produce logical diagrams (formal schematic visualizations) such as migration scenarios.

Stakeholders of structures, such as customers and employees who will use the structure or interested parties who are not domain specialists prefer to see the result of a change translated by way of an artist impression (realistic photographic images). They need, without having to think deeply, to get an instant perception of atmosphere, emotion and feeling.

Managers, analysts, designers, builders and vendors often use drawings to work out scenarios, or to make a choice between alternative options, or in understanding operations, because ultimately they have proven to be effective.

In the following paragraphs we will explore which type of architecture visualization is suitable for different target groups; first of all we will explain the difference between visualizations and architecture visualizations.

5.3.2 Visualizations versus architecture visualizations

Dragon1 differentiates between common visualizations and architecture visualizations. Common visualizations often lack accurate focus of how something is put together or how something operates. Common visualizations lack supportive power in taking decisions. Common visualization often accommodate a mixture of different levels of abstraction, levels of consideration and moments of consideration and therefore lack the ability to be separated. When creating a common visualization, we often forget to notice these aspects.

It is more useful to an architect when he creates architecture visualizations instead of common visualizations. These architecture visualizations assist decision makers to make better choices and architects to communicate more effectively about matters requiring decisions.

An architecture visualization requires elements that illustrate building, administrative, or informatic solutions that include principles and rationales, otherwise it is not a proper architecture visualization.

An optimal architecture visualization is recognizable by its ability to show the total scope of a structure in a transparent way and shows what is or is not in scope of the structure and its environment.

- *An architecture visualization is a visualization that shows which entities, such as concepts, style elements, components, objects and products are part of a total concept, that is used for the architecture design of a structure. An architecture visualization is the result of a conceptual translation made by the architect.*

  An architecture visualization shows which qualitative advantages are included as part of a chosen solution, by concepts, rules and maintenance mechanisms (principles). Hereby the relationship is determined from solutions to ambitions, strategic presumptions, objectives, requirements and preconditions posed by the client and stakeholders involved in the realization of a structure.

There are a number of generic reasons to create an architecture visualization of a structure, or entity such as a (part of an) enterprise, a solution or a concept:

1. **Analysis & Design of a decomposition of a structure in concepts with a delivered quality and output of these concepts.**
   An architecture visualization is created in order to provide insight into the concepts or subconcepts used in a current situation of an enterprise, solution or concept. For that matter, it shows what should be created taking into account future enterprise requirements, solution or concept strategies. Also, how well these concepts fit together and how well they induce quality results. The insight and overview created by architecture visualizations assist in the making of decisions and the choice of different and alternative concepts. Architecture visualization assists in analyzing the impact of change and the ability to choose the best alternative to enhance a structure.

2. **Analysis & Design of the decomposition of concepts in elements and components.**
   An architecture visualization is created to clearly show how well current concepts are realized and implemented at a logical level in elements and at a physical level in components, objects and technical products. Here, the architect puts forward a decomposition of components, whilst providing clarity in function and technique of components and elements. The insight and overview brought by architecture visualizations assist in the making of decisions regarding the choice of an alternative and/or enhanced realization or implementation of a concept. Architecture visualization assists in analyzing the impact of change and the ability to choose the best alternative to enhance elements and components.

3. **Analysis & Design of concept principles and rules.**
   An architecture visualization is created to provide a clear insight into how well current concepts work or how well future concepts must work in relation to strategy and set requirements. Architecture visualizations bring forward operative images, i.e. concept principles and rules. The insight and overview brought by architecture visualizations assist in the making of decisions regarding the choice of an alternative and/or enhanced concept operation (principles).

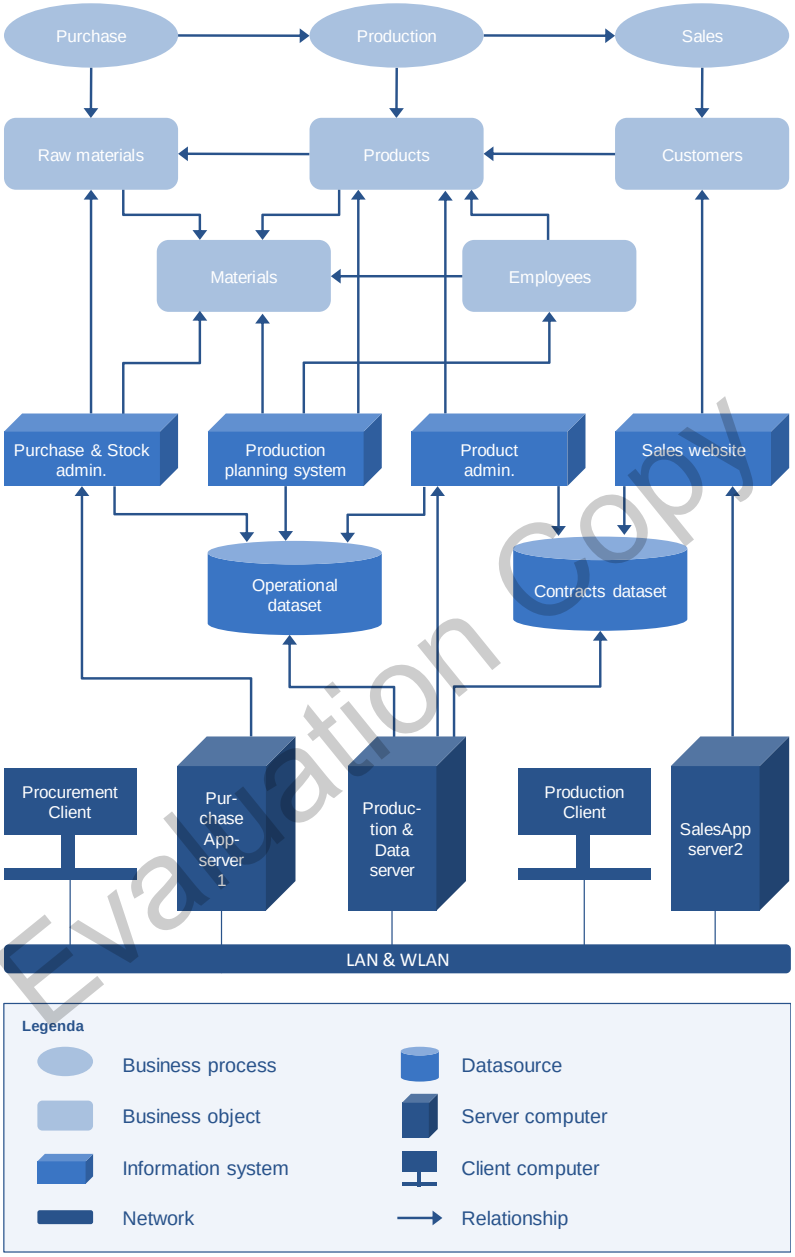

Figure 5.3. Example of a common enterprise visualization.

Figure 5.3 explains how the structure and dependencies of important entities must be shown, but not the actual architecture in these visualization. Also, no insight is

provided about how business administrative, informative and technological concepts are constructed by logical functional elements, physical technical components and objects in this visualization.

In order to create high-quality architecture visualizations, a thorough knowledge of the types of concepts becomes a necessary precondition. Usually the architect possesses adequate knowledge of concepts or concept definitions, showing which entities create style elements and components of a concept. But the architect also requires a thorough knowledge of the effect of concepts in an architecture design, in order to include them effectively. Without this information, the architect is incapable of creating an architecture visualization, only a mere visualization of the relationship between elements and components.

Figure 5.4 shows however a full enterprise architecture. Suppose, a model is created of an enterprise's activities – this is called an business model. An business model, tuned to different target groups can, by including certain specific information, then be graphically presented (visualized) in different ways and as such present other perspectives of the same activities. For instance: by creation of a design sketch, structure vision, and a drawn-up theme map or blueprint diagram. Note: A theme map is in fact the synonym for an aspect vision.

Depending on the kind of target group and the kind of decision that needs to be made, or for that matter efficacious behavior, one type of architectural visualization will be preferred over another, without the danger of co-termination. Further in this chapter when we come to create an architectural visualization and architectural view layout, we will elaborate on the concept of impact behavior.

5.3.3 Types of architecture visualizations

Dragon1 distinguishes different types of architecture visualizations, differentiating visualization techniques and visualization subject-matters.

With regards to the Visualization Framework model, Figure 5.5, Dragon1 recognizes 24 types of architecture visualizations, which enable the architect to fine-tune the way by which he visualizes subject-matters.

At this stage it is not yet about subject-matter information, as this will be covered later on when we are elaborating on aspects of modeling, overviews and perspectives, while referring to the kind of information contained in visualization.

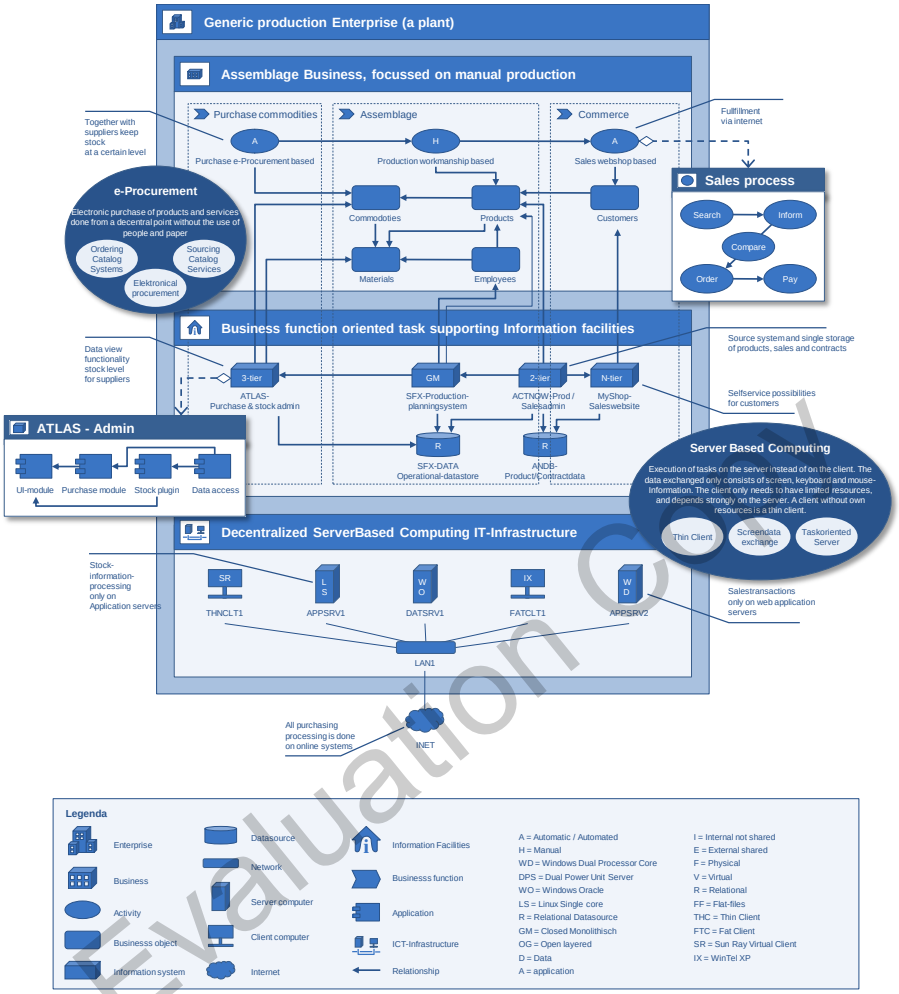

Figure 5.4. Example of an architecture visualization by way of an enterprise diagram.

Dragon1 model – Visualization Framework

	Sketch (Informal) (Inconsistent)	Drawing (Detailed Informal) (Inconsistent)	Diagrams (Detail formeel) (Consistent)	(Photografic) Images (Detailed Informal) (Inconsistent)
Environment / Building locations	Location sketch Environment sketch (Workplace) sketch (User)space sketch	Environment drawing Space drawing Workplace drawing Userspace drawing	Context diagram Workplace diagram Userspace diagram	Location image Environment image (Workplace) image (User)space image
Structural Core / Essence	(total) Concept sketch Subconcepts Structural sketch	(total) Concept drawing Subconcept Structural drawing Cross section	(total) Concept diagram Structural vision ERD	(total) Conceptual image Structural image Collage Persona
Static aspect / Component	Aspect sketch (risk-, security-, quality-, etc…)	Aspect drawing (risk-, security-, quality-, etc…)	Aspect diagram (risk-, security-, quality-, etc…)	Aspect image (risk, security, quality-, etc…)
Situational aspect / Component behavior	Scenario sketch Situation sketch Principle sketch	Story board (Scenario drawing) Situation(detail)drawing Principle(detail)drawing	Story board (Scenario diagram) Situation diagram Principle diagram	Atmosphere image Cartoon Situational image Principal image
A Whole (Assemblage)	Design sketch Building sketch	Design drawing Assemblage drawing Building drawing Layer drawing	Blueprint Assemblage diagram Building diagram Layer diagram	Artist Impression Aarchitecture photo
Building process	Road map sketch ERA sketch Plateau sketch Program sketch Process sketch	Road map drawing Process drawing	Road map diagram Process diagram	Building board Process image
	Inspiration (Emotion, Sphere, Experience, Feeling)	Creation (Financial decision making)	Creation / Control (Item)	Communication (Emotion, Sphere, Experience, Feeling)

Figure 5.5. Visualization Framework.

The **Visualization framework model** distinguishes four principal types of visualization-techniques:

- Sketches
- Drawings
- Diagrams
- Images

In addition to the afore-mentioned, Dragon1 distinguishes six different types of subject-matters, onto which visualizations generally focus within the context of architecture:

- The environment or context
- Core and essence of the structure as a whole
- Aspect or theme of the structure
- The operational aspects (or part) of the structure
- Decomposition of a structure
- The way a structure is designed and implemented.

Visualization by itself, purely the creation of a model, view or perspective stands independently from the contents of the model, view or perspective. Visualization is all about a certain technique, the use and application of visualization instruments released onto certain subject-matter, such as: a whole, part of a whole, a static something, a dynamic something or a process.

Of course, not every single one of the twenty-four types of visualizations is required to follow the route to creating an architecture. The visualization framework model does however serve the purpose of being a valuable checklist. Enabling the architect to visualize on the basis of a choice of subject-matter.

In order to be able to visualize properly, it is always important to realize what we are looking at. Are we looking at the 'whole' or 'part of the whole'? Is the part we are looking at a complete part, i.e. a building block? Or, is it a part of a part of a whole, i.e. a fragment?

What we often notice is that making choices or impact analyses are often based on over-simplified visualizations. However, change often proves to be more resilient than expected. Therefore, by being more explicit and clear about the visualization of the scope of a structure, as well as the environment, the visualization becomes more prominent in its usefulness regarding choices and impact analyses.

In this chapter four principal types of visualizations are elaborated on which are: sketch, drawing, diagram and image – each visualization shown in this chapter contains brief examples of commonly used views.

A view is a specific 'target group oriented visualization' of a model, which is offered as specially presented information. In fact a view is a special sub-model; a part of a model which is visualized with an emphasis on special entity characteristics. Note: When different views are placed together and made to fit to form one whole, it is called a perspective.

5.3.4 Architecture visualization aspects

Each visualization becomes a graphic representation of a reproduction of a model, view or perspective of an entity, such as a structure. The characteristics which will come to order here will therefore also see a recurrence of the characteristics of perspectives, models and systems. This will enable the architect to create an accurate perspective and model, which he is required to draw up prior to making an architecture visualization.

What are the characteristical properties of architecture visualizations, which always need to be recognizable? As a visualization is always marked by many aspects.

The architect is required, with each visualization, to explicitly define the following characteristics.

Feature	Description
Target and target groups	Target and target group/stakeholders (the viewers) of the visualization.
Professional background	Professional background of the viewers, indicative to the way they look at an object.
Subject-matter query	What is the query that is being visualized? How severe is a problem? Why is something a problem? When and how severe will be the manifestation of impact?
Solution	What is the solution (course) being evaluated? What are the client's ideas or preferences regarding a solution?
Assumptions	What are the strategic assumptions or fundamentals that must be taken into account with a query or solution?

Ambitions	Which are an enterprise's higher aims, which must be linked to the query or solution?
Preconditions	What are the preconditions (external requirements) to a solution from a stakeholders' perspective?
Strategy	What are the client and stakeholders' strategy, vision and preferences (documented references)?
Quality aspects	Which quality aspects are important or must be visualized? Openness, complexity, flexibility...
Setting	Background setting of application or presentation of a visualization: is it in relation to a design assignment, a random question, unsolicited or is it with regards to a visualization in reference to previous discussions?
(Types) context(s) or environment (s)	Is the visualization used within or outside its original context? From which perspective is a structure being observed – from within or from the outside?
Total concept	What is the (total) concept behind a visualization or what needs to be shown in this respect? Which concepts are to be shown? Are these existing or new concepts?
(Types) communication message(s)	What is the communication message of the visualization?
Link	How (good) can the relationship be made to fit between strategic information, architecture information and configuration-information regarding the visualization?
Impact behavior	What is the impact behavior that's expected from the visualization?
Decisions	For which solutions will the visualization be used?

Scope	Scope – What are we looking at? What can and cannot be seen? Which entities does the architect include in the background, which are not part of a structure, building block or building block scope? What is (in) the scope of the structure?
Preference solution (course)	Sometimes we see solutions which are carried by the client, or solutions which have already been discussed.
Used visualization principles	Dominance-principle, Proportion-principle, Symmetric-principle Vicinity-principle, viewing window-principle
Eye catchers and viewing windows	What should attract the eye in the first instance? What will be emphasized? Viewing windows and eye catchers.
TO-BE reality principles	Which reality principles should be included in a solution?
Design principles	Which informatic and business administrative design principles in a contextual sense need to be taken into account?
Owner/client	Who is the owner/client of the systems and domains shown in the visualizations?
Design decisions and design criteria	Which design decisions and design criteria have to be incorporated?
Class and type	Used entities: class, type and sort in visualizations
Consideration level	Used consideration level(s)
Moment of consideration	Used moment(s) of consideration
Consideration situation	Used consideration situation(s), the situations in the IT sector and human sector.
Clarity	How clear and proper is the decomposition of requirements, functions, concepts, elements, components and technical

	products in the visualization?
Architecture style	Which (style) elements, components, objects and technical products are included in a visualization?
Quality aspects	The quality aspects which an architect can gather from the need, requirements and assumptions and quality aspects that can be realized by concepts in an architecture.
(Types of) author(s)	Who produced the visualization?
(Types of) view points	What is especially in focus - which expert aspects attract special attention? What is expected and what is intended?
(Types of) perspective(s) /view(s)	Which questions of expertise are actually answered? And what was actually intended?
(Types of) visualization(s)	Have structure, behavior or information been modeled? Will decomposition and relational dependence be mixed?
(Types of) model(s)	Which parts of systems, domains and entities will be emphasized?
Structural decomposition	(Types of) structure(s), building parts, building blocks and building bricks.
Design	What is the (type of) subject in the visualization? Are we watching out for certain problem or a solution (course)?
Environment	What is (present in) the environment of a structure?
Abstraction level(s)	Which abstraction level(s) are used?
Status	Which status and progress information is used?
Levels	Which levels are applied within the visualization? Back office, middle office, front office. How will on every level, environment,

	structure and solution recur?
Legend(s)	Legends of symbols and indicators and definitions of used entities within the visualization.
Document management features	Title, name, distribution, publication, management and process information (author, version, ….).
References	Which documents are used as input for the visualization and in which documents is this visualization used?
Patterns and metaphors	Which patterns and metaphors are used to present information or certain situations or circumstances?

Figure 5.6. Aspects of architecture visualizations.

The better the aspects of existing visualizations can be expressed or brought to the surface, the less a viewer of a visualization has to guess what he is looking at. This way a situation can be prevented where someone does not understand the essence of a visualization, or interprets what he sees as an impendence. Therefore: prepare as much and as well as possible all aspects shown in figure 5.6 in every visualization.

Special attention must be given to prevent that different visualizations are mixed together: for instance one single conceptual, two logical and three physical visualizations drawn into one. Or, abstraction levels and moments of consideration mixed together, which would make a visualization less understandable quickly. A visualization becomes clearer when these aspects are separated and as improved independent part-visualizations reintroduced as a whole.

It also happens frequently that processes are given function-based names, or applications are given nonsensical title names and infrastructure components are given technical names – all of them together in one single visualization. As such it is imperative to raise the nomenclature of the parts in a visualization to the same abstract level, as this will make it easier for the viewer to comprehend.

Similarly, it often happens that the provision of names and entity characterization labeling in a visualization, lack uniqueness or are unequivocally reference-able. It then becomes necessary to provide all entities with consistent analogue

differentiating names and characterization. Characterization can be realized in the form of a word-logo or image-logo.

5.3.5 Sketches

Architects use sketches in order to structure and arrange the vision, ideas and thoughts from himself and other people. A sketch should however not be used to clarify text, but to visualize, concretize and structure thought. Various types of sketches can be applicable: a concept sketch, a principle sketch, a situation sketch or a structure sketch.

Because a sketch provides a rough image they imply that more choices are available. Even when certain aspects of a structure or environment are unclear, a sketch could easily leave one part indeterminate and zoom in on the part that has been defined.

An architect uses sketches to entice and encourage decision makers to make a decision. A sketch is also used to take stock of fundamental requirements, preconditions, strategic assumptions, targets, concepts and principles. Sketches have an important role in establishing appropriate requirements.

- *A sketch is an informal not-established visualization, meant to visualize ideas and thought.*

The concept sketch and principle sketch

An architect will often use sketches of concepts as a quick and simple way to project the advantages of a concept when applied to an enterprise, and how in essence a concept works (principle) and basically which elements and components constitute a concept.

A concept sketch or principle sketch is simple in construction, incorporates a few eye-catching images and presents the scope and contours of a problem or solution. The context, cause, ideas and thought behind a concept are not fully described.

Figure 5.7 presents a concept sketch as used in practice, describing the full support given to two processes consisting of five activities. Here the interdependence of the two processes regarding servers becomes very clear. Frequently business managers only realize how interdependent business processes to IT, after they have seen such sketches.

Because of the speed by which sketches are produced, different sketches can be created for the same solution. Sketches are frequently produced by hand in one

color, with a pencil, crayons or charcoal – subsequently it is called a charcoal sketch and it should never be omitted from any architecture project.

A concept sketch is to all intents and purposes a visualization, a technique used to visualize a concept.

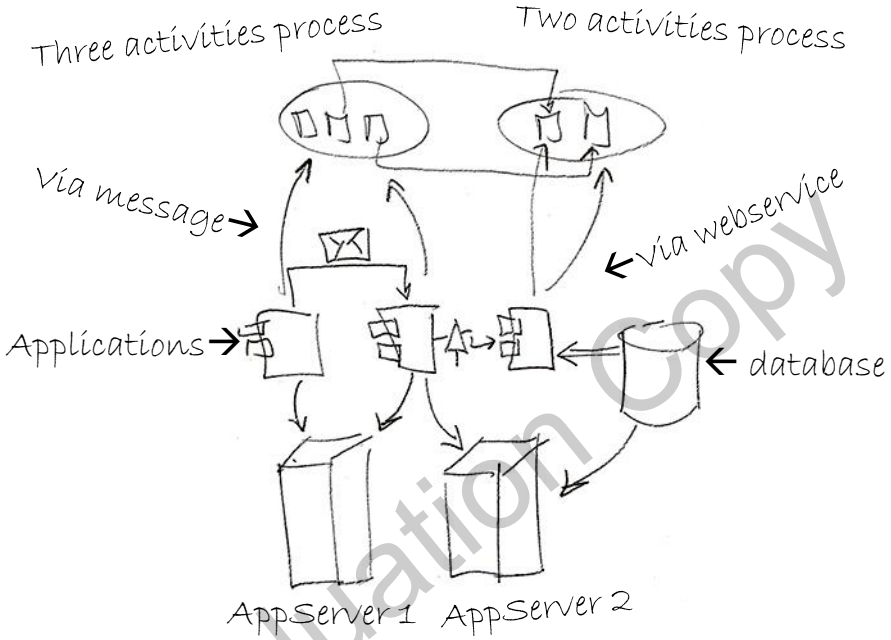

Figure 5.7. Example of a concept sketch.

The design sketch

If a sketch enables the most important design choices of a structure to become understandable the sketch has effectively become a design perspective and as such it will be labeled 'the design sketch'- it is a 'how' sketch.

When the architect and client and/or the most important stakeholders wish to determine which parts of a structure constitutes large building parts, it is advisable to concretize all thought into, for instance a sketch of the total concept.

Figure 5.8. Design sketch of a user's perspective of a total concept.

Figure 5.8 shows a copy of a design sketch. The most important components are visualized, being the concepts of the total concept, whereby archetypes, symbols and metaphors were used. The chosen concepts shown in the copy double as eye-catching aspects of, for instance, a renewal program.

The situation sketch

The least produced sketch is a situation sketch. It describes which characteristic situations could present themselves in the course of designing and realizing a structure and what it should be resilient against.

A situation sketch shows how a problem could be solved in a non-definitive and general way, supported by a realistic course of action, starting from a strategic point aimed at a quality aspect.

Due to the drawing style used in making a situation sketch, people would obtain little or no rights from a visualization in this format. It is in fact no more than a reconnaissance of possibilities and the norm is often that different situation sketches are made at the same starting point.

When photographs or detail drawings are used to make a situation sketch, the advantage is that what is shown is usually concrete. The disadvantage is that people usually expect to derive rights from what is shown.

The symbols in the sketch are elements with an element name and they are often used as metaphors. Situation sketches are meant to serve the decision making process by members of the board, managers and other decision makers.

Figure 5.9 Example of a situation sketch.

Figure 5.9 shows a customer ordering a book on internet. The book can be viewed on screen and in the event the customer wants it, he will be served via a video link of an online book store and order it. The sketch only concretizes ideas but provides little detail.

The structure sketch

A structure sketch is the sketch most often used as it indicates which parts constitute a solution and which sequence can be taken to arrive at a solution.

Figure 5.10 shows which parts constitute the main elements of a solution. We can clearly recognize the three target groups entering an enterprise via a portal. The example in Figure 5.10 is a sketch, because it concretizes ideas. It is not made in support of anything or as an explanation to a text.

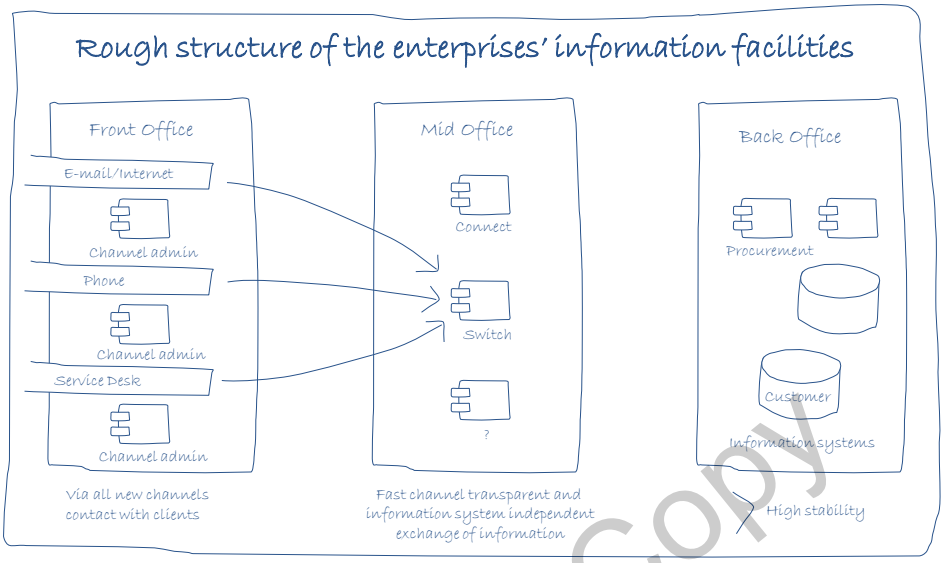

Figure 5.10 Structure sketch of a domain model.

This example sketch was made during a workshop where the participants transferred their thoughts onto a sketch. When a structure sketch displays someone's point of view Dragon1 calls it the 'vision of structure' by this person.

The eye catching parts of the visualization are called eye-catchers. The eye is immediately caught by the contents.

Sketches recur in architecture products, such as a program of requirements, a business case, a conceptual architecture design and a temporary architecture design.

5.3.6 Drawings

An architect will use drawings to illustrate a situation, scenario, moment, story or text. A drawing visualizes, with some detail and clarity, a text. Note: The synonym for drawing is map or infographic.

Dragon1 recognizes different types of drawings, which stand analogue to sketches. The architect produces drawings in support of a story or fragment of a story, which stands in contrast to a sketch.

Suppose, the following is mentioned in a text about communication between applications: 'a customer is served by a staff member who uses software applications. The software application depends on two server computers, which contain other

applications used by different people. The applications are interconnected via the internet, however the servers are not interconnected. The applications make use of each other's database.

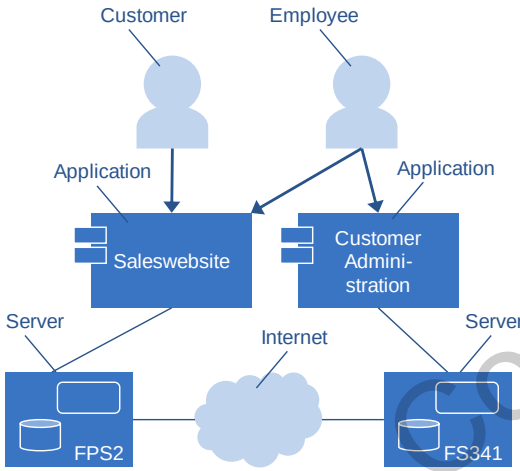

Figure 5.11. Example of an application communication concept drawing.

A reader's power of perception is especially challenged when he is confronted by abstract texts, but with the assistance of a drawing the information will be easier to absorb. Often a drawing proves to be more accurate than a text, and as such assures that textual mistakes can be detected.

The drawing makes use of an architecture annotation technique as it is customary to place under or in a form, the name of the entity being represented and to clarify the position, role or technical form of the whole by an annotation.

Annotations can be left out in the case of an experienced viewer or specialist who is used to working with concept drawings.

It is strongly recommended to make a drawing of the most important points for all plans that do not include a design document. Well known as a rule of thumb says that for every 1000 sentences there are up on average four mistakes. Plans that are expensive to make would benefit greatly from architecture drawings.

Structure drawings

A drawing that reflects the largest part of a structure or solution is called a structure drawing. The structure drawing is the image of a type of visualization and the structure drawing is the image of a type of perspective. Note: The moment this

drawing reflects the vision of someone regarding the division of part-systems and subdomains, Dragon1 refers to this vision as a structure vision.

Figure 5.12. Example of a structure drawing.

Figure 5.12 shows an example of an information facility structure. It shows domain boundaries and the types of entities that constitute a domain.

- *A drawing or illustration is a visual clarification, illustration or explanation of a fragment, connected to a story. A drawing consists of more detail than a sketch and is more accurate than a sketch.*

Situation drawing and principle drawing

A situation drawing displays in an almost definitive way what is expected regarding a solution (course) or it sometimes displays the final solution in order to realize quality and quantity of a strategic starting point.

Because a situation drawing includes many details, it is possible for developers and vendors to make estimations to produce and implement a design.

Situation drawings are made with a thin pencil, fine-liner or by computer. It is a schematic drawing incorporating strict rules regarding the use of symbols and the relationship between symbols.

A situation drawing can be considered a context model (use-case) whereby the use of a certain functionality stands central regarding the solution of a certain role. The symbols used for the visualization of a situation drawing are often physical or digital components or technical products. Consequently they are by nature often very technically correct.

Figure 5.13. Example of a situation drawing.

Figure 5.13 shows an example of a situation drawing: during contact with the customer the staff assistant has to use information regarding customer history which includes product purchase records and product use.

This information enables the staff assistant to be more customer-oriented. The shown drawing contains a lot of details and functions, making it suitable as an explanation to a story. Although the drawing is informal, it remains a situation drawing.

A situation drawing has additional added management value, when it shows a combination of work activities, human dimension and an IT instrument to inform the world at large. Management becomes as such enabled to link with the added business value of an IT instrument.

Situations are often characterized by the imagery of cartoons, sticky messages and humor in order to keep the image alive in the memory of the viewer.

Figure 5.14 Example of a situation drawing in the form of a cartoon.

Figure 5.14 is an example of a cartoon (situation drawing) of a self service concept principle using the medium of an information terminal.

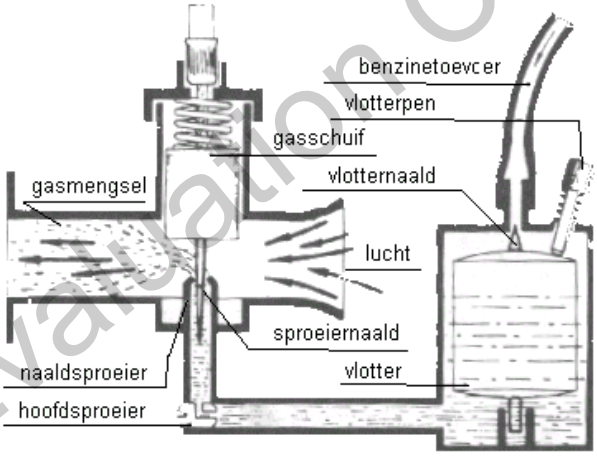

Figure 5.15 Example of a principle drawing of a carburetor.

The principle drawing in Figure 5.15 shows the principle of how a carburetor works.

A drawing is like the sketch an informal visualization that should not be used as part of a contract due to the inaccuracy of the visualization. As a matter of fact a lot of emotional (design) decisions and not financial decisions are taken on the basis of sketches and drawings.

Drawings are often created in PowerPoint or with color pencils. The problem with drawings and sketches is that the use of symbols for certain entities is not or hardly

possible to enforce. Consequently, Dragon1 provides a generic set of sketch, drawing, diagram and image forming, which count as archetype symbols for the most frequently used 250 entities such as process, server, actor, enterprise, information supply and IT infrastructure, as part of the SPACE-architecture style. These generic forms or symbols are divided over five of the most frequently used architectural areas: enterprise, management, business, information supply and IT infrastructure.

5.3.7 Diagrams

When an architect needs to accurately describe or specify the decomposition or operation of a structure or a part thereof, the most appropriate visualization technique is a diagram, also called a scheme.

There are methods, implicitly equalizing diagrams and schematics with visualizations. Dragon1 provides the architect with more freedom by offering the possibility to choose different types of visualizations instead of just the diagram technique.

Dragon1 distinguishes different types of diagrams for different types of subjects which could be visualized by way of a diagram. Dragon1 has for the purpose of visualizing the construction and operation of models adopted UML model diagrams as the standard. It concerns the following diagrams:

- Use-Case diagram (i.e. the context model)
- Class (model) diagram
- Object (decomposition model) diagram
- Component (decomposition model) diagram

All of the above diagrams show hierarchy of classification, objects and components. From an analogue perspective, Dragon1 also distinguishes diagrams pertaining to:

- Concept (decomposition model) diagram
- Element (decomposition model) diagram
- Technical product (decomposition model) diagram
- Artifact (decomposition model) diagram.

These diagrams are characterized by the fact that they visualize only one type of entity with regards to interrelationships and hierarchy, for instance, an application component model or business process model.

Dragon1 also recognizes, next to decomposition diagrams, configuration diagrams, which show different types of entities. We hereby consider the business configuration model, which shows both, application components and business processes together with possibly another forty or more types of entity.

Another type of diagram is the dynamic behavior diagram, which shows a certain processes in operations, as follows:

- Collaboration model diagram
- Sequence model diagram
- Situation model diagram
- Process flow diagram
- Event flow model diagram

These diagrams show; which sequential steps are taken by a system, in which situation a system could arrive as a result of change and as a result of which event a system reacts.

The last type of diagrams used in a visualization framework are information diagrams – examples are:

- Fishbone diagram
- Centric diagram
- Framework diagram
- Mind-map diagram

These diagrams are distinguishable or characterized by a certain composition of information and data within a visualization.

> *A diagram or scheme is a formal, constructed, often consistent visualization in which rules, roles and constraints apply as to how forms between entities of models or perspectives are represented.*

A structure diagram

A structure diagram could be used to visualize in a simple way scope, impact and reach of change in an enterprise.

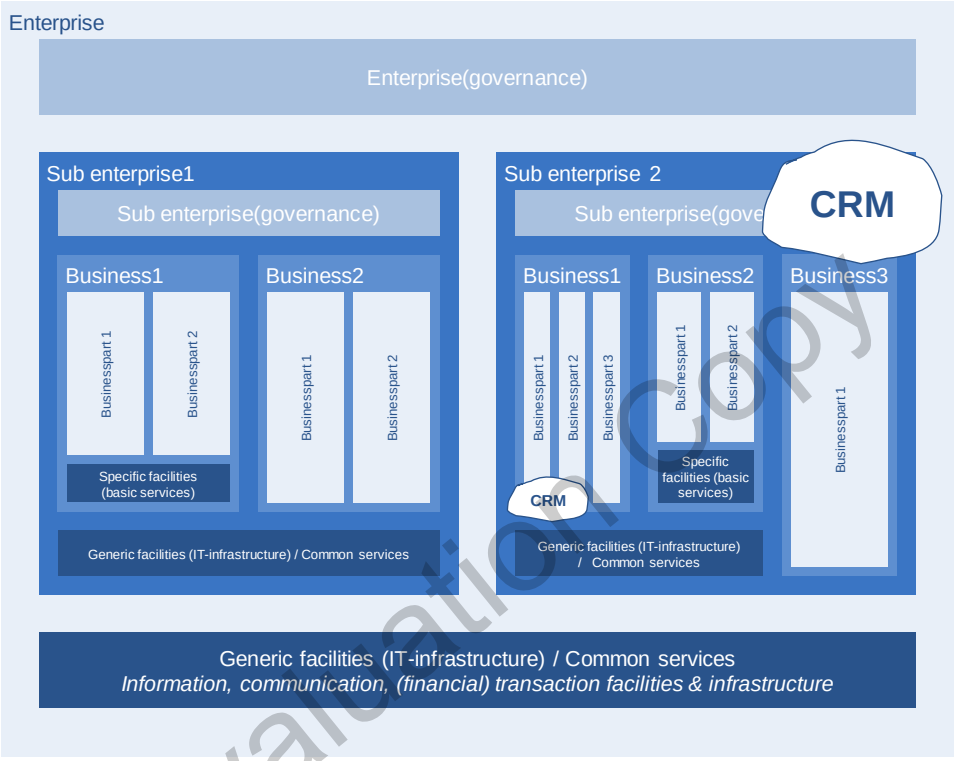

Figure 5.16. Example of a structure diagram.

Figure 5.16 accurately displays enterprise domain boundaries. The structure diagram provides an example of the possible resistance experienced when trespassing over a domain boundary.

The visualization shows a renewal program taking place inside business function 1 of business 1 of enterprise branch 2. This program has no direct impact on other business functions within the enterprise branch 2. The effect of the CRM program gets however bigger when management decides that the CRM program is applicable on a higher enterprise domain branch level. CRM becomes more effective as its application gets applied in a wider sense and this principle is visualized by the above structure diagram.

A principle diagram

A principle diagram is a formal visualization displaying the operations of a structure, concept or phenomenon, which gets sometimes coupled to a design need or impact. In other words: the results that are achieved by its operations.

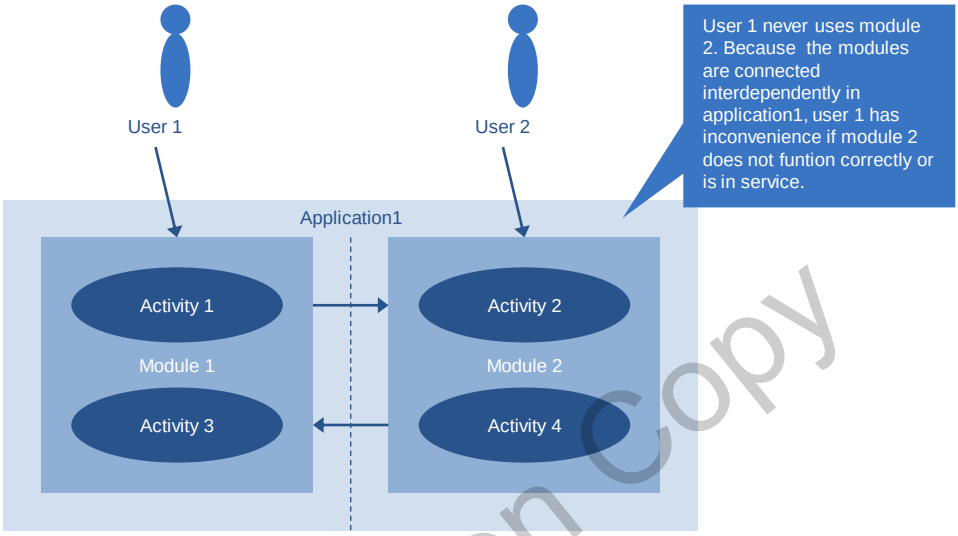

Figure 5.17. Example of a principle diagram.

Figure 5.17 accurately displays the relationship between elements. This differentiates the visualization from a drawing.

We notice in the visualization the recurrence of the principle of interdependence regarding modular application, by which the two modules show an undesired high interdependence between each other. If one module fails the other is directly affected. This principle often occurs but is in essence still insufficiently visualized.

Component diagram

An architect will often produce a formal element diagram, showing as accurate as possible the relationships between logical and functional elements. This also applies to an object diagram and technical product diagram.

A component diagram shows how many components there are, how they are interrelated and which components the architect used to make a decomposition. Also shown in the background are components, which although they are not part of the scope, make it clear how the solution is an integrated part of the total.

This diagram notation is a standard for ultimate interchangeability and reusability in an enterprise.

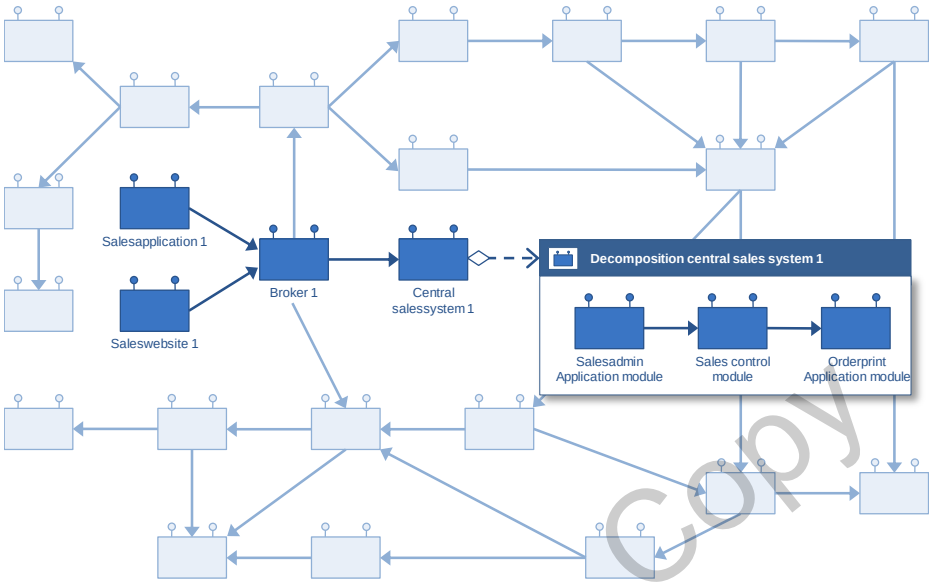

Figure 5.18. Component model diagram.

The architectural product diagram is different from other diagrams in such a way that it enables the architect to visualize which architecture products, product parts, models, views and perspectives were made or planned for a structure and for which target group the architecture products were meant. Increasingly architectural product diagrams are made at a baseline or a quick scan.

5.3.8 Photographic images

Many fundamentally complex and strategic innovations in an enterprise change collectively and it is not always clear to everyone what the end result will be. Especially with regards to customers and personnel a broad support base must be created when major renewals are planned. This is one reason why an architect uses photographic images of a structure, building part or building block. A photographic image closely resembles the real object allowing everyone to get an idea of what it will eventually look like.

Dragon1 distinguishes between the following types of photographic images:

- Architecture photographs
- Artist impressions

- Collages as shown in Figure 5.19

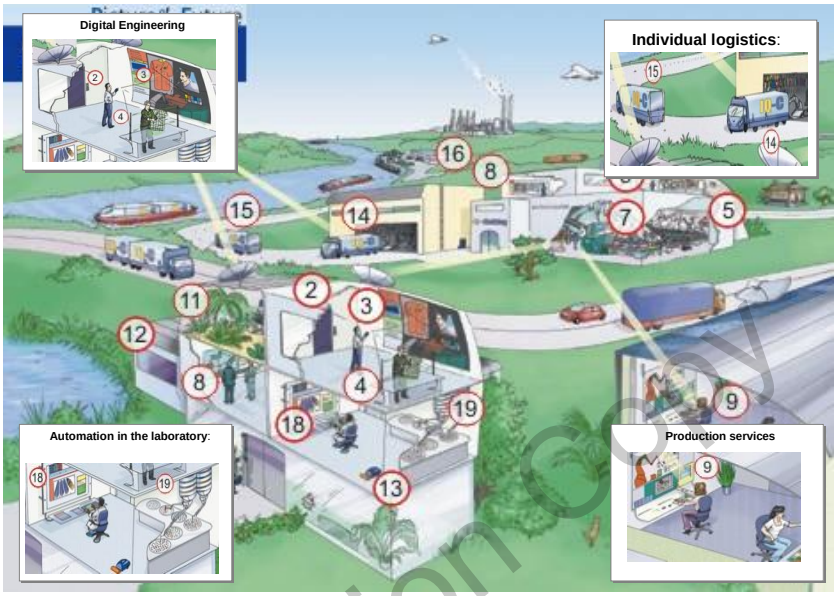

Figure 5.19. Example of a photographic image.

> *A photographic image is a truthful, detailed and very accurate reproduction of the actual object. Only by looking intently at a real photograph and a mock up picture the difference becomes visible.*

The situational image

An image of a situation shows in a definitive way and in precise details which solutions are pursued. It contains atmosphere, perception, emotion and feelings and it often consists of photographic collages, photographic visualizations or cartoons. Because of the details that are portrayed people derive rights from what they see.

Figure 5.20. Example of a situational image.

Situational images are meant for a wide audience, but especially for stakeholders, such as personnel and customers.

5.3.9 Situations, spaces and locations

Dragon1 differentiates between the visualization of a situation, visualization of a location and visualization of a space. Simply put, an enterprise is the working space of personnel, where products and services are used by customers. Consequently use situations and work situations exist where people work and buy, use space and work space exist for work and sales and use locations and work locations exist to work and sell. Architects specifically design facilities and infrastructures for these places. The architect usually creates three types of visualizations:

1. Visualizations to assist stakeholders to determine the most appropriate requirements regarding the elements of a solution or a structure.
2. Visualizations to assist stakeholders to make a choice in alternative component solutions of a design in line with, or derived from concepts and elements.
3. Visualizations to assist stakeholders to make technical product choices in support of the realization of a structure, derived from components.

The architect may chose to make collages of situations in order to make it easier to make a difficult decision. These collages are combinations of images,

diagrams, drawings and sketches in which situations of work and utilization are shown for the sake of the client, the architect himself, a manager, an engineer or another stakeholder, who has to make a decision or make a choice on certain requirements.

Use Situation

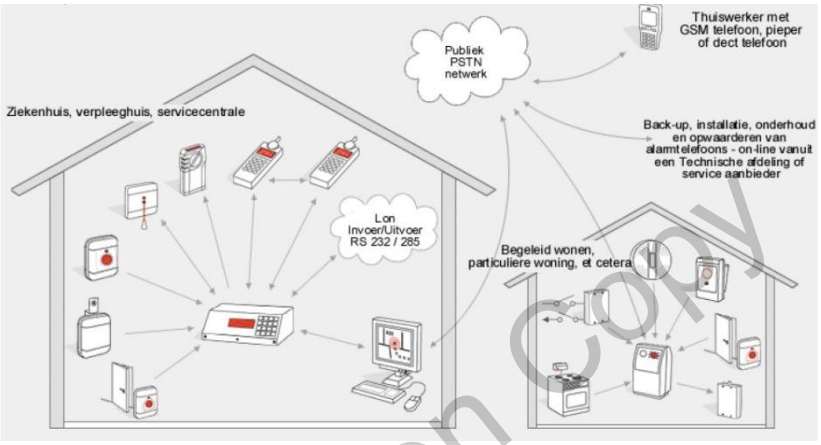

Figure 5.21. Example of a use situation.

With the sketch in figure 5.21 (source: Handy Guide) of a use situation, the architect is able to gauge what it is that client and stakeholders exactly want, are allowed and can do in terms of functional specification.

People are now able to imagine what the new situation will look like - a photographic image of a person in a certain situation and scenario, a combination of events, triggers, times, places, activities and locations and environment.

Working space

Figure 5.22. Example of the drawing of a work space.

The example drawing of Figure 5.22 of a work space shows how angry customers are brushed aside. The architect can use the cartoon to show office workers how they treat angry clients now and that customer care can be improved. Another example of a cartoon showing an image of application the architect is able to test the possibility of fitting another high-performance work space in a current work environment

Use location

Figure 5.23. Example of a photographic image of a use location.

This photographic image from Figure 5.23 enables the architect to test early on in the project to test with users and workers, regarding functional specification, whether new customer processes and worker processes can be integrated within the hotel, restaurant and bar industry.

He will create miracles by including visualizations, with photographic images, to show users, customers and workers in a clear way how to find solutions to a problem, in a user or work situation. This way it soon becomes clear to a wide audience how something is or becomes a problem. The situation should always include a life event, environment, location, spot, act, task or activity.

Although we often aim to achieve independence of space and location, often there remains a dependence in space and location regarding certain use-case activities. Consequently, the architect must often make collages of situations in which problems present themselves or will present themselves in the future and how this problem can be solved or avoided.

It pays the architect to think in terms of scenarios, such as in collages of situations at various moments in time, covering certain events and triggers.

5.3.10 Views

Frameworks, Visions, Landscapes and Blueprints

It is important for an architect to make a correct difference between frameworks, landscapes, structure visions and blueprints. These are different views on subjects in the enterprise.

Dragon1 makes a big difference between the structure of an entity or structure and the architecture of an entity or structure. The structure makes visible the interrelationship and interdependence of entities in a system such as a structure. The architecture makes visible the presence and relationship of concepts of a structure.

If we combine structure and architecture of an enterprise with the four types of views, we have eight views with which we can visualize in a short time, in outline, the most important facets of an enterprise architecture for an consideration period, such as AS-IS, TO-BE or Envision.

With a framework the architect visualizes in fact a meta model. With a structure vision, with a landscape and with a blueprint the architect visualizes actually a model that fits within the meta model.

An architect makes clear towards the board and management, for example with views such as a logical AS-IS enterprise structure framework and a physical TO-BE

solution structure blueprint, how the enterprise looks now and to what extent the to be realized solution fits within the current enterprise.

Framework views

Frameworks are classification views or views of entity relation meta models. They make clear to a subject, system or structure how something is organized, what kind of entity types are in it. A framework is basically a cabinet with compartments where something can be put in, or boxes for reasons left empty. For an enterprise it is valuable to prepare a framework for communicating certain subjects.

Frameworks are to sketch, to draw, to map and to imagine photographically. The visualization of the same framework can therefore look very different.

Dragon1 recognizes two major classes of frameworks: structure frameworks and architecture frameworks.

An architecture framework is a view that gives insight into the architectures that are recognized in a structure. An AS-IS enterprise architecture framework provides a clear view, which architectures are recognized in the enterprise, who 'owns' an architecture and the progress with documentation, including agreeing architecture policy. An architecture framework is, in fact, equivalent to structure a conceptual framework.

A structure framework is a view showing the main elements of concepts of a subject, system or structure. A TO-BE business structure framework shows which business domains, business functions, business processes in an enterprise are identified for the future. The fact that in the business domains, functions and processes are identified used and reused, again must result from the business architecture framework. A structure framework that visualises elements is a logical structure framework. A structure framework that visualises components and objects, is a physical structure framework. A structure framework that visualises technical products, is an implementation structure framework.

Creating a framework is usually preceded by the creation of a structure, landscape or blueprint. A framework is in fact the box of parts that will be used in a structure, landscape or blueprint.

Using color or words in a framework to highlight that certain things are unknown, not well organized or clash with other things, makes this view more suitable for taking decisions.

Vision views

Visions are images that someone sees before him. This may not be the truth or reality, but may be the current image or the future image of someone. Dragon1 recognizes structure vision and architecture vision.

A structure vision is the image that someone has for the current and/or future situation regarding an entity, system, concept or structure at the enterprise. For example, the TO-BE process structure vision is a view showing overview and insight into the recognition, division, grouping and placement of processes in the enterprise.

Analogous to the structure vision, Dragon1 recognizes the architecture vision. An enterprise architecture vision is for example a view with which we see at a glance the most important concepts in an enterprise. An enterprise structure vision must contain the most significant elements that are present in these concepts. In for example a process architecture vision contain the concepts that follow from the processes, which are recognized by a structure vision process.

Structure views are not rigid or sharp visuals. They are mostly sketches which illustrate an dynamic, a transition and an renewal, or just highlight a pressing issue.

Landscape views

Landscapes are views that show for a very limited set of types of entities, the presence, group, division and placement. Dragon1 recognizes two main types of landscapes: architecture landscapes and structure landscapes.

An architecture landscape shows where and how the concepts in a given system, such as a structure, are present. For example, an AS-IS application architecture landscape is a view that makes clear what types of application concepts and application principles are currently present.

Adjacent to the applications are data sets, interfaces and links. In an application architecture landscape it is therefore natural to see even those concepts. In a landscape it is also a good use to visualize the things that are visible in the area, but do not have the attention, to lighten, in the color gray or faded. So the architect can make the difference between background and foreground, which increases the value of information and the communication message of the visualization.

A structure landscape is different from a landscape architecture. A structure landscape shows where and which elements, related to concepts, are present in a system such as a structure. For example a TO-BE information structure landscape shows where the physical and digital information in the enterprise is present and how it manifests itself.

As the name 'landscape' suggests, sketches, drawings and photographic images of a landscape often show an undulating surface where different concepts and elements are present.

Like frameworks, we also recognize logical, physical and implementation landscapes as landscapes.

Blueprint views

Blueprints are detail views of entity relation models, including the attributes. Blueprints are sharp cut-outs that can be used in projects as a guide for realizing a business process or information system. The amount of information on a blueprint entails that not just anyone can see through the complexity.

Dragon1 recognizes two main types of blueprints: architecture blueprints and structure blueprints.

An architecture blueprint shows in detail all the concepts that are in a certain area in the enterprise, to what other concepts in the area they are related, and to which they depend. Also indicated, who is the owner and manager, and how well the concepts are or should be implemented. For example, an enterprise architecture blueprint shows, other than an enterprise architecture vision, all concepts that have been identified for an enterprise. How these concepts are visualized (a wire figure or a photographic image of the concept) depends on the type of visualization is chosen to visualize the blueprint.

A structure blueprint shows issues in detail, like the architecture blueprint, but includes all the elements or components from a concept. A logical AS-IS IT infrastructure blueprint is a view that shows all existing and recognized elements of the existing IT infrastructure and includes ownership, status, condition.

Blueprints often have a high information density and complexity. Sometimes blueprints are also very useful to show to the board and management, and to give them an idea of the complexity and uncontrollability of the situation. Something that

will help to directors making a choice to make everything a lot easier, reduce costs, and get a higher quality.

5.3.11 Bringing views under policy

In order to make the switch from dreaming working with architecture, to real working with architecture, views should be determined and approved by a client, such as a CIO. After this, the views must be placed under policy.

This means that a policy document specifying the approved version of a view, and how the view is required to use. If this step is forgotten in the architectural process, architecture remains a paper tiger, and thus never comes to life.

5.4 Questions

After studying this chapter the reader should be able to answer the following questions.

5.1. Which are Dragon1's eight concepts of the 'way of representing'?

5.2. What kinds of architecture visualizations are there?

5.3. What are sorts of structure models?

5.4. What are viewpoints, views and perspectives?

5.5. How does the architect create an architecture poster?

Chapter 6

Dragon1 Glossary of Terms

The new enterprise architecture language.

6.1 Introduction

This chapter 'Dragon1 Glossary Of Terms' provides an overview of the definitions of the most important architectural terms used in this book, consisting of more than 2000 terms of the the Dragon1 Open Method. Because the glossary of terms is continuously expanding and improving, it is not completed in this book. The entire list of terms belonging to Dragon1 can be found on http://wiki.dragon1.org.

The importance of a common glossary of terms is underwritten by everyone, but the glossary of terms has to be sufficiently original, consistent, and mature in order not to cause limitations. These are the conditions of the glossary of terms where innovation is required and Dragon1 provides adequate space.

In order for others to discuss architecture and architecture subjects efficiently and effectively, it is of importance to use the same language and terms. Of course outside Dragon1 architectural terms are defined and provided, but as far as Dragon1 is concerned it is insufficient and too little. Many terms turn out to be colloquialisms rather than appropriately discussed terms, which stand up to scrutiny.

Additionally, it is apparent to redefine a number of terms because their current meaning form a blockage to the further development of Dragon1, as opposed to existing architecture methods. This specific nature of Dragon1 is inextricably connected to terms or as we call it: the Glosssary of Terms.

All defined terms shown in the list are official Dragon1 terms. The English term is leading throughout Dragon1 and the Dutch term has been derived from English. In the Netherlands the discipline of enterprise architecture allows English and Dutch terms to be exchanged. Consequently, the glossary of terms table includes a Dutch – American/English translation table.

The terms in the list are provided with stipulative definitions. A stipulative definition means that a specific description of the content of the term is given, of which the description of the word (term) is composed by Dragon1. Consequently, this deviates from the user's definition of a word (term). In other words, it is the used meaning of a word in a certain culture or discipline. These definitions are always true within the scope of Dragon1.

From a Dragon1 perspective, in the current architecture discipline different architectural terms have been used for many years as a container term and are somewhat diluted in terms of the original definition, function and usage. Dragon1 focuses with terms on stipulative redefinition. We sharpen the architectureal terms and return architectural terms to their original meaning, which is mostly building architecture. The Dragon1 terms and definitions of terms deviate from the common definition used in the discipline of enterprise architecture. This is only done this way to provide visual enterprise architecture with more added-value within the confines of an enterprise.

In the coming years more research is required for the further development of Dragon1 architectural terms, including consistency, translation and usage thereof in literature. This includes the glossary of terms list, which will undergo continuous development. Therefore keep yourself informed of the latest developments.

Below are the definitions of the most important terms from this text book.

Activity

An activity is a set of works, tasks or actions.

Architecture

Dragon1 - Architecture, in the sense of the discipline of architecture, is the art and science of a planned, function-oriented and integrated design and realization of a structure that is mostly durable and future-proof.

Dragon1 - Architecture, in the sense of architecture of a structure, is the coherent whole of decorative, operative and constructive concepts that is or will be applied to a structure.

IEEE1471 - Architecture is the fundamental organization of a system embodied in its components, their relationships to each other and to the environment and the principles guiding its design and evolution. (2011)

Architecture Design

An architecture design is a schematic representation of a plan, or representation of a solution to a design issue, for instance, a structure that supports the needs of stakeholders. In an architecture design, complex or difficult design issues are solved with the help of concept and principles. Hereby functionality and form were designed separately as well as construction, decoration and operation of the solution. Altogether this makes a design, an architecture design.

An architecture design is a design of a total concept for a structure which consists of an analysis of a current configuration, space, environment, location, and situation of an existing system and a design of a future configuration, space, location and situation. The configuration, space, location and situation are made visible and controllable with descriptions and visualizations of entities at a conceptual, logical, and physical level. In an architecture design the translation is made from concepts to elements, from elements to components and objects, and

from components and objects to technical products.

Architecture visualization An architecture visualization is a visualization that shows which entities, such as concepts, style elements, components, objects and products are part of a total concept, that is used for the architecture design of a structure. An architecture visualization is the result of a conceptual translation made by the architect.

An architecture visualization shows which qualitative advantages are included as part of a chosen solution, by concepts, rules and maintenance mechanisms (principles). Hereby the relationship is determined from solutions to ambitions, strategic presumptions, objectives, requirements and preconditions posed by the client and stakeholders involved in the realization of a structure.

Artist Impression An artist impression (artist's impression is correct english) is the impression that an architect has of the whole of situations that are in or around a structure that can, should or will occur. In these situations the functions, capabilities and forms of the structure are made visible and understandable to a wide audience.

Artifact An artifact is (often an unique) structure, element, component, object or product recognized by people, which is designed and made with great skill.

Blueprint A blueprint is a detailed construction or assembly plan. A blueprint is a building or assembly view aimed at a structure on a two dimensional plane.

Building Block A building block consists of a total of entities meant to be used again (repetitively) in a structure.

Business A business is an organization where people collaborate in business processes in which they create and deliver goods, products and services in order to collectively achieve results or to achieve goals concerning services and products.

Business Function A business function is a set of elements and components such as activities that are focused on achieving a common functional purpose. For instance, the business activity sales, with sales staff, sales processes and sales information systems.

Business Process A business process is a structured time-sequential set of activities, with ownership, usually within the scope of a business function, but always across organizational units.

Capability A capability is a degree of competence. It is an ability or skill that an entity often acquires or develops by, through or with another entity, knowledge, experience or skill.

Component A component is a more specific technical physical (composite) entity of a system, with it's own characteristics, properties and/or behavior. A component is one part of a concept, structure or phenomenon.

Concept A concept is an approach, an abstraction of a solution or an idea. A concept is nothing more than a direction for a solution.

Concept Sketch A concept sketch is an informal not-established visualization of a concept, meant to visualize ideas and thought behind the concept, its core structure, usage and

	way working.
Design Assignment	A assignment for an architecture design is an assignment in which the architect is awarded a contract with sufficient mandate to make an architecture design of a structure or a part thereof. The client is often an owner of a structure in an existing situation or as owner of real estate of a new structure. The contract must also state how quickly a project must be completed and how much the implementation of the design may cost.
Design Rule	An agreement between two or more parties on a particular method of design or way of creating a design.
Design Sketch	A design sketch is an informal not-established visualization, meant to visualize ideas and thought of a total concept.
Domain	A domain is defined and an imaginary area of responsibilities, tasks, roles, responsibilities and ownership.
Element	An element is a more generic, logical functional (composite) component of a system. An element is one part of a concept, structure or phenomenon.
Style Element	A style element is an element, a logical functional entity, that is used more than once in the design and construction of a structure and is therefore characteristic to that structure.
Enterprise	An enterprise, such as a care institution or a bank-insurer is a whole of businesses that, perform in a certain unique way a range (products and services) for the market, by which they take consciously a responsible risk.
Enterprise Architecture	Enterprise Architecture, in the sense of the discipline architecture, is the art and science of a planned, function-oriented and integrated design and realization of an enterprise structure that is mostly durable and future-proof. Enterprise Architecture, in the sense of architecture of a structure, is the coherent set of decorative, operative and constructive concepts that is or will be applied to a structure.
Enterprise architecture blueprint	An enterprise blueprint is a detailed construction or assembly plan for a whole enterprise or integral part of it. An enterprise blueprint is a building or assembly view aimed at an enterprise-structure on a two dimensional plane.
Enterprise architecture framework	An organization and classification of architectures, concepts and principles.
Entity	An entity is something that is recognized or acknowledged. An entity has its own identity and can be distinguished from other entities. An entity has attributes that provide the entity with an identity.
Environment	The environment around a system in which other systems interact with eachother
Facility	A facility is an entity, device or system, usually with a specialized function that is made available to a large group of users, to users in their basic needs.
Fragment	A fragment is an incomplete section that is part of a structure.

Function	A function is the execution of tasks by an entity. A function dictates what an entity is able to do. A function is the total of the results that can be provided by an element or component. A function does not have a physical form or implementation form, a function does not yet possess a physical identify or implementation identity – a form, however always has one or more functions.
Identity	The distinctive appearance of an entity
Infrastructure	An infrastructure is a set of fixed services (facilities) for common use, where users have access, but do not have ownership.
Master plan	A master plan is an overall planning of all activities to ultimately achieve an realized architecture design.
Meta model	A metamodel is an abstraction of a model. A metamodel is a set of entity classes and relationships between entity classes that specifies which entities and relationships from a model that may be made based on the metamodel. A meta model is basically a template for a model.
Mission	A mission is the set of tasks of an enterprise, a business or business function.
Model	A model is a collection of related entities with relationships between those entities, including aspects of time, place and action aspects.
Need	A need is the consciousness or desire for something missing. It is the sense of necessary completion of requirements for certain work to be performed.
Object	An object is a more specific technical abstract (composite) component of a system characteristics, properties and/or behavior. An object is thus a part of a structure, concept or phenomenon.
Organization	An organization denotes a system to have certain facilities or features. Colloqually organization is often used as a synonym for enterprise. The best way to use this term is: 'The organization of ...' for instance a foundation, business, process, business, etc ...
Owner/Client	An owner-client is the person who gives the architecture design assignment to a person or organization in order to have a structure or solution designed and realized using architecture.
Perspective	A perspective is the combination of views where the views are shown in mutual relationship relative to each other. Views are therefore sometimes deformed in order to form a whole. A perspective is the way in which something such as a part or aspect of a structure, presents itself to a viewer from a certain point of view. A perspective is to be represented textual and visual.
Principle	A principle is the enforced way by which an entity operates, happens or takes place with a certain effect or produced result with regards to a given context.
Architecture Principle	An architecture principle is a concept or design principle that is devoid of all contexts or a design principle that can be used always and everywhere in a particular system. An architecture principle is an universal principle that is

	applied in the holistic enterprise and therefore a comprehensive statement about the 'construction' of the enterprise.
Concept Principle	A concept principle is a principle that shows the way a concept has been put together, how it works or what has materialized in the concept, from which a certain result is always produced or a certain effect is noticeable.
Design Principle	A design principle is a principle that is applicable to the environment of a structure and as such influences the structure. The architect must take these principles consciously into account when designing and realizing a structure.
Reality Principle	A reality principle is a principle that states what the enforced way of working is within a realized system (or part of that system). In this case, a reality principle denotes a principle that applies to the current situation in an enterprise.
Principle Drawing	A principle drawing is a drawing that explains the enforced way an entity works.
Pattern	A repeatable arrangement of configuration of items meant to solve a specific problem.
Production	Production is the process that results in a quantity of delivered products, goods or services.
Program of requirements	A program of requirements (PoR) is a document describing the needs and demands of stakeholders in relation to the forms and functions of a structure or a solution that are prioritized and grouped by performance and quality aspects.
Performance Requirement	A performance requirement is something that definitively should come into existence with respect to a quality aspect. It is a necessary feature, function or performance to be delivered from something. It is urgently required, as opposed to a wish.
Quality Requirement	A performance requirement is something that definitively should come into existence with respect to a quality aspect. It is a necessary feature, function or quality to be delivered from something.
Phenomenon	A phenomenon is a common occuring but striking event. A phenomenon is an event that is observable. Phenomenon literally means 'observable'.
Precondition	A precondition is an additional requirement, or condition which is necessary for achieving a goal. It is an external requirement of other stakeholders. A minimum requirement by third parties, which at least must be met.
Rule	A rule is an agreement concerning the relationship between two or more entities. If the agreement is not abided by, a sanction comes into force consisting of a variety of consequences. The severity of the sanctional consequences determines whether it becomes a rule, a regulation, a direction or a law.
Situation	A situation is an interpretation by an observer of an event on the basis of the beholder standards and values.
Stakeholder	Persons or organizations with a certain interest in the development of a solution or have an interest in a particular situation.

Strategic starting point

A strategic starting point is a declaration that makes it clear via its Board of Directors what the enterprise stands for, where it is coming from, and what it regards to be its basic and fundamental principles. A strategic starting point makes it clear what to take into account regarding, environment, situation, issues, trends, development and information in relation to execution of its core activities.

A strategic starting point gives direction and imposes borders. As such, it provides structure, tranquility, rhythm and space. Strategic starting points have a great impact in setting and implementing strategy. A strategic starting point resonates environment, ambition, objectives, intention, critical success factors, mission, vision, philosophy, culture as well as the identity of the enterprise. Strategic starting points are preferably formulated as statements.

Strategy

Strategy is the set of fixed (long term) goals, tasks and tasks of an enterprise. This includes the planned integrated and function oriented way how the goals should be achieved and key tasks to be performed. Also, the named and identified targets, efforts and results and the allocation of necessary resources such as people and their skills, resources and their functions needed to achieve the goals, are part of the strategy.

Structure

A structure is every construction of a physical material such as for instance wood, or digital material such as automated process support, more often than not, furnished for operational or decorative purposes. The automated process support at its end destination is directly or indirectly supported on the basis of, a platform where, services or functionality are delivered locally or remotely. A structure consists of constructive, operative and decorative concepts, elements, components, objects and products.

Structure Vision

A structure vision in the context of the strategy is a function & form layout plan for a part or the whole of a structure.

System

A system is a set of components that work together to achieve a common goal.

Total concept

A total concept is the consistent total of collaborating concepts, which the architect uses as a basis for the architecture design. In fact, we could soon see a hundred of these concepts and more, as well as large parts or components of the architecture.

View

A view is the interpretation of what a person sees or finds important by virtue of his function, concerns, knowledge and skills. A view is a subset of a model.

Vision

A vision is a certain view on one or more themes that are important for being able to perform tasks and achieve missions.

Visual Enterprise Architecture

Visual enterprise architecture is the art and science of creating architecture designs for structures with a focus on visualizing different models, views and perspectives. This enables clients, stakeholders and engineers to increase management control, have insight and overview, exercise support and conduct the realization of the design, functionality, performance and quality of structures.

Literature References

The following publications were used as background materials for the method and this book:

1. Adams, Manfred, HUF HAUS, www.huf-haus.com
2. Bandlamudi, Ethos, Volume 22, issue 4, Dec., 1994, 460-493
3. Berg, M. van den ea, Wegwijzer voor methoden bij enterprise-architectuur, 2009, Van Haren Publishing
4. Berkel, Ben van, Erasmusbrug, 1996, UNStudio
5. Bishop, Franklin, Cartoons Tekenen, 2006, Librero
6. Bos, Drs R. ten, Business Process Redesign, Bedrijfskunde Tijdschrift voor modern management, jaargang 69, 1997, nummer 1
7. Ching, Francis D.K., Design Drawing, 1998, Van Nostrand Reinhold, New York
8. Cole, Emily, De taal van de architectuur, 2003, Librero
9. Dabner, David, Design en Lay-out, 2004, Librero
10. Engel, Henk ea, Wat is architectuur? Architectuur-theoretische verkenningen, 2007, Uitgeverij SUN Amsterdam
11. Gamma, Erich, Design Patterns, 1995, Addison-Wesley
12. Grit, R., Informatie-management, 2002, Wolters-Noordhoff
13. Haaften, Adriaan van, Uit de hand, schetsvaardigheden in de bouwkundestudie, 1997, Faculteit der Bouwkunde, Delft
14. Hamel, G. en Prahalad, C., "The Core Competence of the Corporation", Harvard Business Review, vol. 68, nr. 3, mei-juni 1990, pagina 79-93
15. Porter, M.E., Activity System Map, What is Strategy? Harvard Business Review, Nov-Dec 1996
16. Herrema, Auke, Principe-illustratie van een polderboezem-systeem, gemaakt in opdracht van Hoogheemraadschap van Delfland.
17. Hilliard, Rich, Rich Hilliard Papers, http://mysite.verizon.net/rfh2/writings/
18. Hoogenboom, Sander, Praktisch modelleren met UML 2.0
19. Huttinga, Ep en Zeeland, Winfried van, Bouwkundig tekenen in de ontwerpfase, 1990 Faculteit der Bouwkunde, Delft
20. Instituut Nederlandse Kwaliteit, INK, 2003, www.ink.nl
21. Jans, Prof E.O.J., Grondslagen van de administratieve organisatie, 2001, Stenfert Kroese
22. Kant, Immanuel, Kritik der Reiner Vernunft / The Critique of Pure Reason, 1781
23. Krebs, Robert E., Encyclopedia of Scientific Principles, Laws, and Theories, 2008
24. Leupen, Bernard, ea, Ontwerp en analyse, 1993, Uitgeverij 010 / Technische Universteit Delft
25. Macaulay, David, The Way Things Work, 2004
26. Martin, James, Information Engineering, 1989
27. McDonough, William, The Almere Principles, 2008
28. Meyer, Han, ea, Het ontwerp van de openbare ruimte, 2006, Uitgeverij SUN Amsterdam
29. Mittelmeijer, Marc ea, Kijk op bedrijfsprocessen, 2007

30. Netten, J. van, Handreiking Functioneel Specificeren, Ministerie van Verkeer en Waterstaat, ExpertiseCentrum Opdrachtgeverschap (RWS, ECO), http://www.verkeerenwaterstaat.nl/kennisplein/page_kennisplein.aspx?DossierURI=&Id=341353
31. Nieuwenhuis, M.A., The Art of Management (the-art.nl), Zeist, 2003-2008
32. Norman,W. Edmund, The Scientific Method, http://www.scientificmethod.com/ 1997
33. OMG Object Management Group, www.bpmn.org, BPMN
34. OMG Object Management Group, www.uml.org, UML
35. ORM, www.orm.net, ORM
36. Phillips, E. Barbara, City Lights - Urban-suburban life in the Global Society, 1996, Oxford University Press, New York
37. Porter, M. E., What is strategy?, 1996, Harvard Business Review, November-December, 61-78, The value chain
38. Rau, Thomas, ONEPLANETARCHITECTURE, http://www.rau.nl/
39. Rijsenbrij, Daan, Architectuur in de Digitale Wereld Rijsenbrij oratie, Radboud Universteit Nijmegen, 1994
40. Rumbaugh, J, Object Oriented Modeling and Design, 1990, Prentice Hall
41. Ruys, Frédérik, http://www.vizualism.com/, Visualism
42. Salden, M.W.R. en Nes, J. van, Bouwkundig tekenen, basisbegrippen, 1997 ThiemeMeulenhoff, Utrecht/Zutphen
43. Sanden, Drs Wim van der, Informatie-architectuur, de infrastructurele benadering, 1997, Panfox
44. Schieman, C.J. ea, Beheersing van bedrijfsprocessen, 2002, Stenfert Kroese
45. Tanenbaum, Andrew, Gestructureerde Computer-architectuur, 1990, Academic Service
46. Tapscott Don, ea, Paradigm Shift, 1992
47. TU Delft, http://www.bouwtechnischtekenen.bk.tudelft.nl/
48. Verma, Surendra, The Little Book of Scientific Principles, Theories & Things, 2005, Reed New Holland
49. Vinci, L. da, The Notebooks of Leonardo Da Vinci - Complete, 1881
50. Vitruvius Pollo, M., De Architectura, 25 VC
51. Wang, Thomas C., Plan and section drawing, 1979 Architectural Press, Londen
52. Wieringa, Roel, ea, Competences of IT Architects, 2009
53. William McDonough & Michael Braungart, Cradle-to-Cradle, Remaking the Way We Make Things, 2002, North Point Press
54. Wilson, E. Bright. An Introduction to Scientific Research, McGraw-Hill, 1952
55. Wurman, Richard Saul, Pionier in informatiearchitectuur, http://www.wurman.com/
56. Zachman, John, The Zachman Framework, 1990, Zachman Institute
57. Zeel, Mo, Architectuur ontwerpen en tekenen, 2008, Librero

We have tried to research as thorougly as possible the rightfull owners of theorieës, models and other aspects of knowledge. If in your opinion a reference is missing, please report this as soon as possible at the Dragon1 Architecture Foundation via info@dragon1.org.

Acknowledgements

Many organizations and persons have contributed the past years to the creation of the Dragon1 Open EA Method. Dragon1 was practiced at various small, medium sized and large organizations to develop it and improve it. We would like to thank everyone for his or her commitment and added value. Without you Dragon1 would have never been there. A few organizations and persons we would like to mention here.

Stakeholders, Reviewers and Ambassadors

Ad Schinkel, Alfred Dijkstra, Anja van Deijk, Daan Rijsenbrij, Erik Douwes, Erik Proper, Ferry Brasz, Frank van Dijk, Frans Breedveld, Frans van den Berg, Fijtse Vos, Frits de Boer, Gerard Deijk, Gerben Hoogeboom, Greg Mulder, Han van Roosmalen, Harmen Groenwold, Henk Jurrien, Henk Morang, Henk Stienstra, Jacob de Boer, Jan Herbrink, Jacques Janssen, Jo Aendenroomer, Jo Sanders, John Pape, Jos Dagelinckx, Leo van Brandwijk, Leon Schurgers, Marijn Driel, Marleen Olde Hartmann, Manuel Herrera, Matt Janssens, Paul Rekveld, Paul Gloudemans, Peter Bloks, Rene van Hove, Richard Uijen, Robert Tros, Ron van Harn, Ronald de Gier, Ruud de Haas, Sjaak Aalten, Ton Eusterbrock, Thijs van Bruchem, Willem Reek.

Organizations where Dragon1 was first practiced

Agis Zorgverzekeringen - Amersfoort, ASR Verzekeringen - Utrecht, Connexxion - Hilversum, Crédit Agricole - Amsterdam, Delta Lloyd - Amsterdam, Gemeente Gouda, Gemeente Roerdalen, Gemeente Steenwijkerland, Gemeente Sneek, Gemeente Maastricht, Holland Casino - Hoofddorp, HWW Zorg - Den Haag, ING Bank - Amsterdam, KPN - Den Haag, OHRA - Arnhem, Sociale Verhuurders Haaglanden - Den Haag, UWV - Amsterdam, Woningstichting Rijnmond - Rotterdam, Vestia - Delft, Ziekenhuis Gelderse Vallei - Ede.

Dragon1 Architecture Foundation Boardmembers

Jan Verwoerd, Chairman
Ad Paauwe, Secretary & Vice Chairman
Alfred Dijkstra, Treasurer